NAVAL POSTGRADUATE SCHOOL

MONTEREY, CALIFORNIA

THESIS

CONTROL SYSTEM OF A THREE DOF SPACECRAFT SIMULATOR BY VECTORABLE THRUSTERS AND CONTROL MOMENT GYROS

by

William D. Price

December 2006

Thesis Advisor: Marcello Romano

Approved for public release; distribution is unlimited

THIS PAGE INTENTIONALLY LEFT BLANK

REPORT DOCUMENTATION PAGE	*Form Approved OMB No. 0704-0188*

1. AGENCY USE ONLY *(Leave blank)*	**2. REPORT DATE** December 2006	**3. REPORT TYPE AND DATES COVERED** Master's Thesis

4. TITLE AND SUBTITLE Control System of a Three DOF Spacecraft Simulator by Vectorable Thrusters and Control Moment Gyros

5. FUNDING NUMBERS

6. AUTHOR William D. Price

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)
Naval Postgraduate School
Monterey, CA 93943-5000

8. PERFORMING ORGANIZATION REPORT NUMBER

9. SPONSORING /MONITORING AGENCY NAME(S) AND ADDRESS(ES)
N/A

10. SPONSORING/MONITORING AGENCY REPORT NUMBER

11. SUPPLEMENTARY NOTES The views expressed in this thesis are those of the author and do not reflect the official policy or position of the Department of Defense or the U.S. Government.

12a. DISTRIBUTION / AVAILABILITY STATEMENT Approved for public release; distribution is unlimited	**12b. DISTRIBUTION CODE** A

13. ABSTRACT

This thesis presents the continued design and system integration of a prototype three Degrees-Of-Freedom (DOF) Spacecraft Simulator used in the Proximity Operations Simulator Facility, as part of the Naval Postgraduate School's Spacecraft Robotics Laboratory, to simulate autonomous guidance, navigation and control (GNC) for spacecraft proximity operations and assembly as part of the Autonomous Multi-Agent Physically Interacting Spacecraft project. Several key enhancements of the spacecraft simulator were made including the integration of onboard sensors, improved electrical distribution system, improved command and data handling system, and the design and integration of vectorable thrusters.

A pair of independently controlled 360 degree vectorable thrusters is now included in the spacecraft simulator. A control system and thruster mapping algorithm were developed to incorporate the translational and rotational control authority that the vectorable thrusters provide with the rotational control authority of the previously developed Miniature Single-Gimbaled Control-Moment-Gyroscope (MSGCMG). Simulation and experimental results are presented to demonstrate the functionality of the prototype AMPHIS vehicle. The work done in developing the prototype vehicle will enable rapid fabrication of additional vehicles to provide essential hardware-in-the-loop experimentation capabilities for evolving control algorithms, sensors and mating mechanisms to be used for autonomous spacecraft assembly.

14. SUBJECT TERMS Autonomous on-orbit spacecraft assembly, fractionated spacecraft, vectorable thrusters, proximity operations, multi-agent, robotic, hardware-in-the-loop test bed	**15. NUMBER OF PAGES** 99
	16. PRICE CODE

17. SECURITY CLASSIFICATION OF REPORT Unclassified	**18. SECURITY CLASSIFICATION OF THIS PAGE** Unclassified	**19. SECURITY CLASSIFICATION OF ABSTRACT** Unclassified	**20. LIMITATION OF ABSTRACT** UL

THIS PAGE INTENTIONALLY LEFT BLANK

CONTROL SYSTEM OF A THREE DOF SPACECRAFT SIMULATOR BY VECTORABLE THRUSTERS AND CONTROL MOMENT GYROS

William D. Price
Lieutenant, United States Navy
B.S., United States Naval Academy, 1999

Submitted in partial fulfillment of the
requirements for the degrees of

MASTER OF SCIENCE IN ASTRONAUTICAL ENGINEERING
and
ASTRONAUTICAL ENGINEER DEGREE

from the

NAVAL POSTGRADUATE SCHOOL
December 2006

Author: William D. Price

Approved by: Marcello Romano
Thesis Advisor

Anthony J. Healey
Chairman, Department of Mechanical and Astronautical Engineering

THIS PAGE INTENTIONALLY LEFT BLANK

ABSTRACT

This thesis presents the continued design and system integration of a prototype three Degrees-Of-Freedom (DOF) Spacecraft Simulator used in the Proximity Operations Simulator Facility, as part of the Naval Postgraduate School's Spacecraft Robotics Laboratory, to simulate autonomous guidance, navigation and control (GNC) for spacecraft proximity operations and assembly as part of the Autonomous Multi-Agent Physically Interacting Spacecraft project. Several key enhancements of the spacecraft simulator were made including the integration of onboard sensors, improved electrical distribution system, improved command and data handling system, and the design and integration of vectorable thrusters.

A pair of independently controlled 360 degree vectorable thrusters is now included in the spacecraft simulator. A control system and thruster mapping algorithm were developed to incorporate the translational and rotational control authority that the vectorable thrusters provide with the rotational control authority of the previously developed Miniature Single-Gimbaled Control-Moment-Gyroscope (MSGCMG). Simulation and experimental results are presented to demonstrate the functionality of the prototype AMPHIS vehicle. The work done in developing the prototype vehicle will enable rapid fabrication of additional vehicles to provide essential hardware-in-the-loop experimentation capabilities for evolving control algorithms, sensors and mating mechanisms to be used for autonomous spacecraft assembly.

THIS PAGE INTENTIONALLY LEFT BLANK

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

THIS PAGE INTENTIONALLY LEFT BLANK

LIST OF ACRONYMS

AMPHIS	-	Autonomous Multi-agent Physically Interacting Spacecraft
NPS	-	Naval Postgraduate School
NASA	-	National Aeronautics and Space Administration
JPL	-	Jet Propulsion Laboratory
DARPA	-	Defense Advance Research Projects Agency
MIT	-	Massachusetts Institute of Technology
SSAG	-	Space Systems Academic Group
SRL	-	Spacecraft Robotics Laboratory
POSF	-	Proximity Operations Simulator Facility
DOF	-	Degrees of Freedom
COTS	-	Commercial Off the Shelf
CMG	-	Control Moment Gyro
MSGCMG	-	Miniature Single Gimbaled Control Moment Gyro
C&DH	-	Command & Data Handling
iGPS	-	Indoor Global Positioning System
GNC	-	Guidance, Navigation and Control
LiDAR	-	Light Detection and Ranging
RS-232	-	Recommended Standard -232
TCP/IP	-	Transmission Control Protocol/Internet Protocol
CPU	-	Central Processing Unit
LEO	-	Low Earth Orbit
I/O	-	Input/Output
USB	-	Universal Serial Bus
KVM	-	Keyboard, Video, Mouse
FOG	-	Fiber Optic Gyro
TTL	-	Transistor-Transistor Logic
LED	-	Light Emitting Diode
DOC	-	Disk on Chip

HDD	-	Hard Disk Drive
IDE	-	Integrated Drive Electronics

ACKNOWLEDGMENTS

The author would like to acknowledge the financial support of the Naval Postgraduate School.

The author would also like to thank the following for their invaluable assistance in the completion of this thesis:

Dr. Marcello Romano for his expertise and guidance throughout the thesis process.

LCDR Blake Eikenberry and LCDR Jason Hall for their friendship and partnership in the developmental process of the AMPHIS test bed.

Captain Dave Friedman and LCDR Tracy Shay for their discoveries of many best-practices in component selection and robotic vehicle construction.

THIS PAGE INTENTIONALLY LEFT BLANK

I. INTRODUCTION

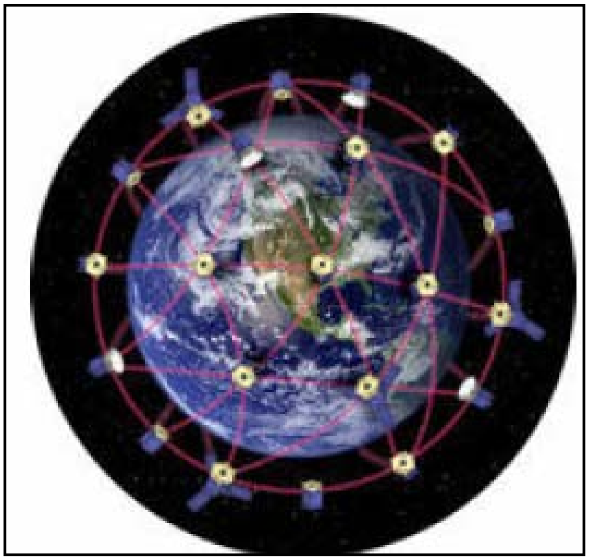


Figure 1. Globally Distributed Fractionated Spacecraft (Ref. [1])

A. MOTIVATION FOR MULT-AGENT SPACECRAFT SYSTEMS

The Autonomous Multi-Agent Physically Interacting Spacecraft (AMPHIS) test bed is designed for the purpose of testing various control and navigation methods that could be used for the autonomous interaction of multiple homogenous fractionated spacecraft network during proximity operations. Current space systems use a one time, single use model for accomplishing a specific mission. These space systems are developed and operate under a high level of risk. The traditional approach to managing these risks is increased design margin and built in redundancy, both of which lead to higher cost.

Fractionated spacecraft systems may offer an architectural diversity approach to manage this risk. They offer greater flexibility, diversification of risk, and spatial distribution. Homogeneous fractionated spacecraft systems are composed of several identical interacting agents that are small-scale replicas of a traditional large spacecraft and are fully capable of functioning independently of each other. No longer would a single, costly rocket launch be needed for all missions; instead, some missions could be launched in multiple smaller vehicles as independent units. Once on orbit, the units would autonomously maneuver and dock to form a single functioning satellite to perform a single mission. (Ref. [2])

B. CURRENT MULTI-AGENT SPACECRAFT SIMULATORS

The task of multiple autonomous spacecraft operating in a coordinating manner requires the development of highly technical control systems. Using smaller, independent spacecraft vehicles requires exploring new sensor and actuator approaches in order to perform all the necessary functions in a small platform. The high cost involved with on orbit testing, along with the need to develop new control systems, leads to the necessity of developing and utilizing multi-agent spacecraft simulators. A shortlist of available simulators is presented here below.

1. MIT Synchronized Position Hold Engage and Reorient Experimental Satellites (SPHERES) Laboratory

The MIT Space Systems Laboratory developed the SPHERES (Synchronized Position Hold Engage and Reorient Experimental Satellites) laboratory environment to provide DARPA, NASA, and other researchers with a replenishable and upgradeable test bed for the validation of high risk control and autonomy technologies used in formation flight and autonomous docking, rendezvous and reconfiguration algorithms. The test bed consists of three small, self-contained vehicles, or "spheres," which can control their relative positions and orientations, and is operable on a 2-D laboratory platform, NASA's KC-135 (shown in Figure 2), and the International Space Station. (Ref. [3])

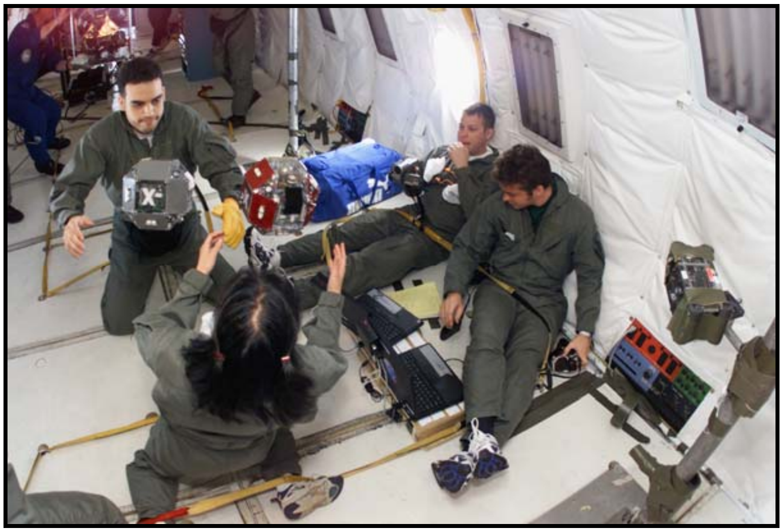


Figure 2. MIT SPHERES Testing on KC-135 (Ref. [3])

2. JPL 6 DOF Formation Control Test bed

JPL's six Degrees of Freedom (DOF) capable Formation Control Test bed (FCT), shown in Figure 3, is designed to support technology demonstration and development of the requisite control algorithms for fractionated spacecraft interaction. Each FCT robotic vehicle consists of a star tracker, a CG balanced attitude platform containing three reaction wheels, 16 1N air thrusters, wireless Ethernet and a compactPCI Bus PowerPC750 flight computer. A linear air bearing, three air pads and a vertical stage provide 6 DOF for each vehicle by 16 3000 PSI air tanks. Each of three robots stands 64.5 inches high with a diameter of 59.5 inches, weighs approximately 358 kg and operates in a 40 foot diameter room. (Ref. [4])

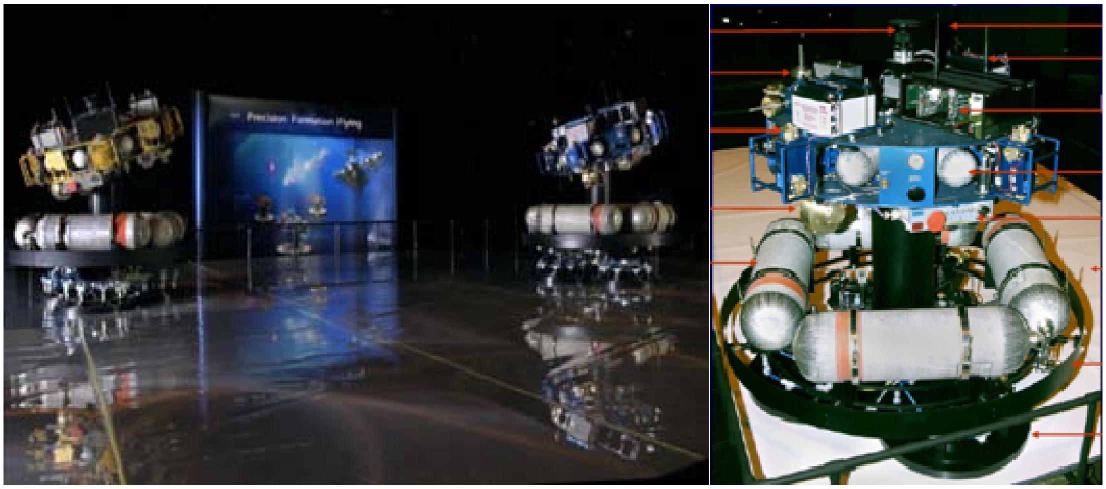


Figure 3. JPL Formation Control Test bed (Ref. [4])

3. NPS SRL Previous Work

The Spacecraft Robotics Laboratory (SRL) at the MAE Department of the Naval Postgraduate School supports the Graduate School of Engineering and Applied Science (GSEAS), the Space Systems Academic Group (SSAG), and conducts research for the Air Force Research Lab (AFRL) (Space Vehicle Directory), Defense Advanced Research Projects Agency (DARPA) (Tactical Technology Office), and various sponsoring agents. The first interacting spacecraft simulator robot project conducted in the SRL at NPS was the Autonomous Docking and Servicing Spacecraft Simulator (AUDASS), shown in Figure 4. (Ref. [5], [6], [7])

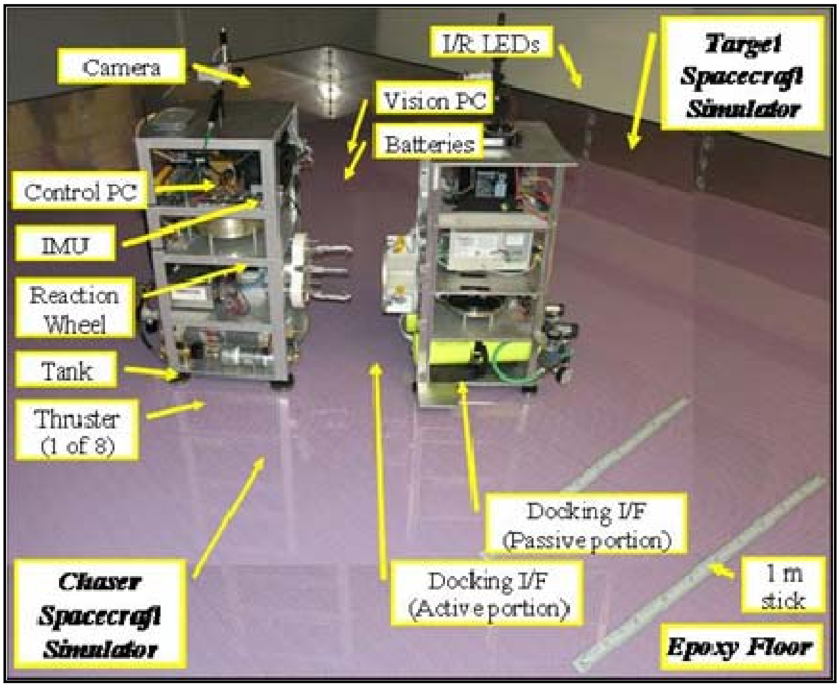

Camera
I/R LEDs
Target Spacecraft Simulator
Vision PC
Batteries
Control PC
IMU
Reaction Wheel
Tank
Thruster (1 of 8)
Docking I/F (Passive portion)
Docking I/F (Active portion)
1 m stick
Chaser Spacecraft Simulator
Epoxy Floor

Figure 4. Autonomous Docking Test bed at the NPS SRL (Ref. [5], [6], [7])

The Proximity Operations Simulator Facility (POSF) was developed to validate Guidance, Navigation, and Control (GNC) approaches for autonomous docking and fractionated spacecraft missions. The facility utilizes computer modeling and simulation with hardware in the loop testing. The facility consists of a 4.9 m x 4.3 m epoxy floor with indoor GPS and floating spacecraft simulators. (Ref. [5], [6], [7], [8])

As a follow on spacecraft simulator to the AUDASS test bed, the AMPHIS test bed has been developed. The benefits of the AMPHIS test bed over the AUDASS test bed is a smaller, more modular design that utilizes more commercial of the shelf (COTS) equipment making it is less expensive platform. These benefits will along for the construction of more simulators and the testing of multiple different sensor, actuator, and controller combinations easily. The previous work on the development of the AMPHIS test bed, shown in Figure 5, consisted of the physical structures, the air system, the electrical system, and the design and development of a Miniature Single Gimbaled Control Moment Gyro (MSGCMG). This initial development of AMPHIS is documented in Ref. [8].

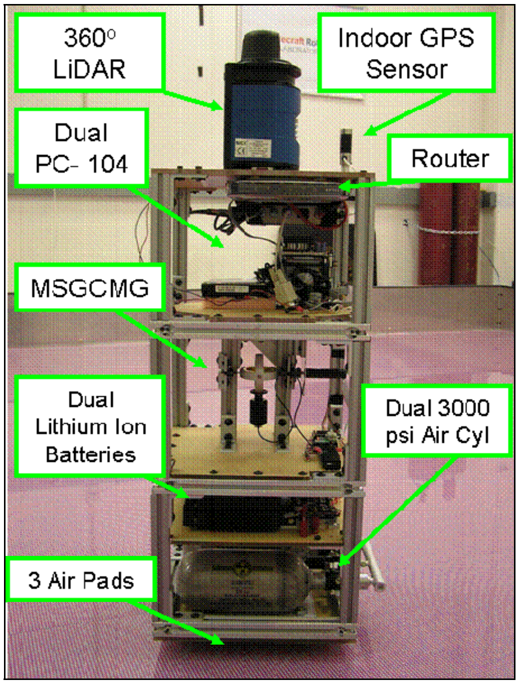


Figure 5. Early AMPHIS Spacecraft Simulator (Ref. [8])

4. Scope of Thesis

The goal of this thesis is to document the further development of the AMPHIS test bed. The new developments made include improvements to air and electrical systems, the addition of onboard sensors, the development of the command and data handling (C&DH) system, and the addition of vectorable thruster actuators.

The improvements that are made to the air and electrical systems have double the endurance of the spacecraft simulator. A fiber optic gyro, a three-axis accelerometer, an iGPS receiver, as well as a LiDAR sensor have all been added to the spacecraft simulator. A pair of vectorable thrusters has also been added, giving the spacecraft simulator translational control authority adding to the attitude control provided by the MSGCMG. All of the necessary software required to integrate these sensor with the C&DH system have also been developed.

In addition to the hardware improvements, a controller and actuator mapping algorithm has been developed. These improvements, along with high level software development and LiDAR integration, discussed in Ref. [9], allow for full system testing of the AMPHIS test bed. Several rendezvous maneuver experiments have been successfully completed. Simulation and experimental results are provided.

II. COMPLETION OF THREE DEGREES-OF-FREEDOM (DOF) PROTOTYPE SPACECRAFT SIMULATOR

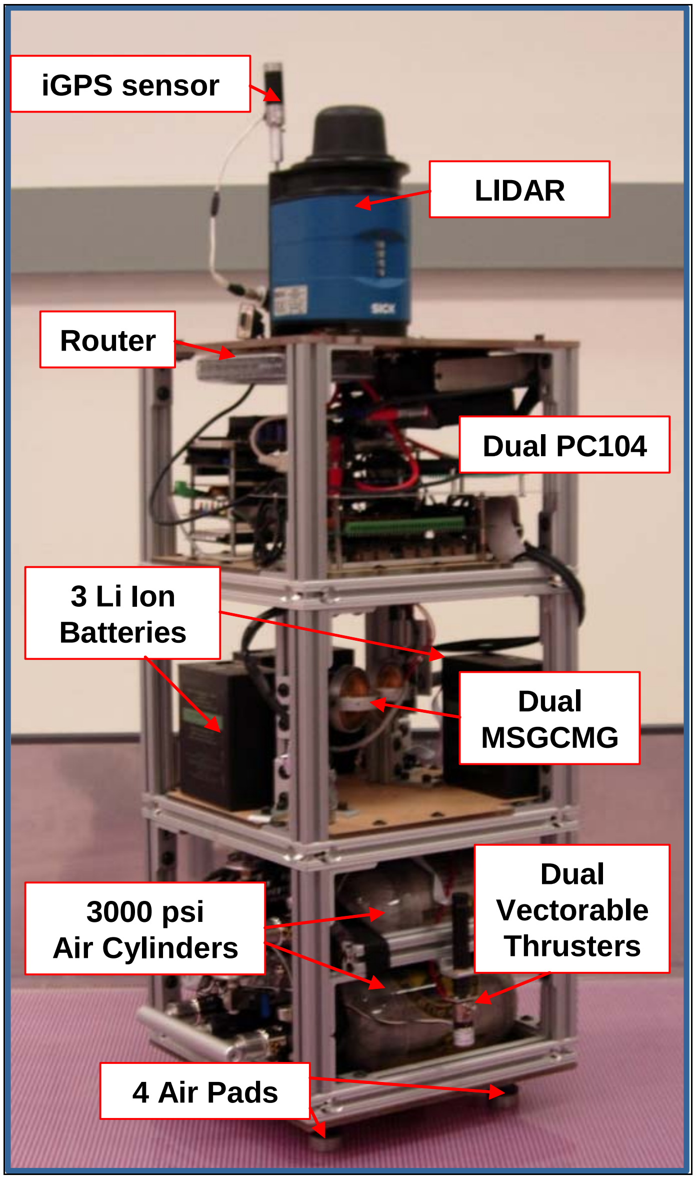


Figure 6. Prototype Spacecraft Simulator for the AMPHIS Test bed in the POSF

A. OVERVIEW OF SPACECRAFT SIMULATOR UPGRADES

A three degree-of-freedom spacecraft simulator, shown in Figure 6, was developed for use in the AMPHIS test bed. Much of the early development of the spacecraft simulator, including structures, initial air and electrical systems, and the design of a Miniature Single Gimbaled Control Moment Gyro (MSGCMG) had been accomplished earlier and are described in detail in Ref. [8]. The following sections describe the remaining work that has been done including improved air and electrical systems, further development of the command and data handling system, the addition of onboard sensors, and the addition of vectorable thrusters.

The overall layout of components in the spacecraft simulator has been drastically altered. This new arrangement allows for the addition of two additional air cylinders, an additional battery, and the addition of another gyroscope to the MSGCMG assembly. These enhancements allow for longer simulation run times as well as increasing the attitude control authority of the MSGCMG. The new overall spacecraft simulator characteristics are depicted in Table 1.

Size	Length and Width	.30 [m]
	Height	.69 [m]
	Mass	37 [kg]
	Moment of Inertia about Z_{ch}	.75 [kg m^2]
Flotation	Source	Air
	Equivalent Storage Capacity @ 21 [MPa] (3000 PSI)	.002 [m^3]
	Operating Pressure	0.35 [MPa] (50 PSI)
Propulsion	Propellant	Air
	Equivalent Storage Capacity @ 21 [MPa] (3000 PSI)	.002 [m^3]
	Operating Pressure, Thrusting	120 PSI
	Cont. Operation (No Thrust Factored)	75 [min]
	Thrust of Each Thruster	.28 [N]
	CMG Max Torque	.334 [Nm]
	CMG Max Ang. Momentum	.049 [Nms]
Electrical & Electronic Subsystem	Battery Type	Lithium-Ion
	Storage Capacity	12 [Ah] @ 28[V]
	Computers	2 PC104 Pentium III
Sensors	Fiber Optic Gyro Bias	±20°/hr
	LiDAR Sensor	Under Development
	Optical Position Validation Sensor	Under Development
	Pseudo-GPS Sensor Accuracy	< .050 [mm]
	Accelerometers Bias Stability	$\pm 8.5x10^{-3}$ g
Docking I/F	Magnetic	Under Development

Table 1. Updated Key Parameters of the AMPHIS Prototype Spacecraft Simulator

B. AIR SUPPLY SYSTEM IMPROVEMENTS

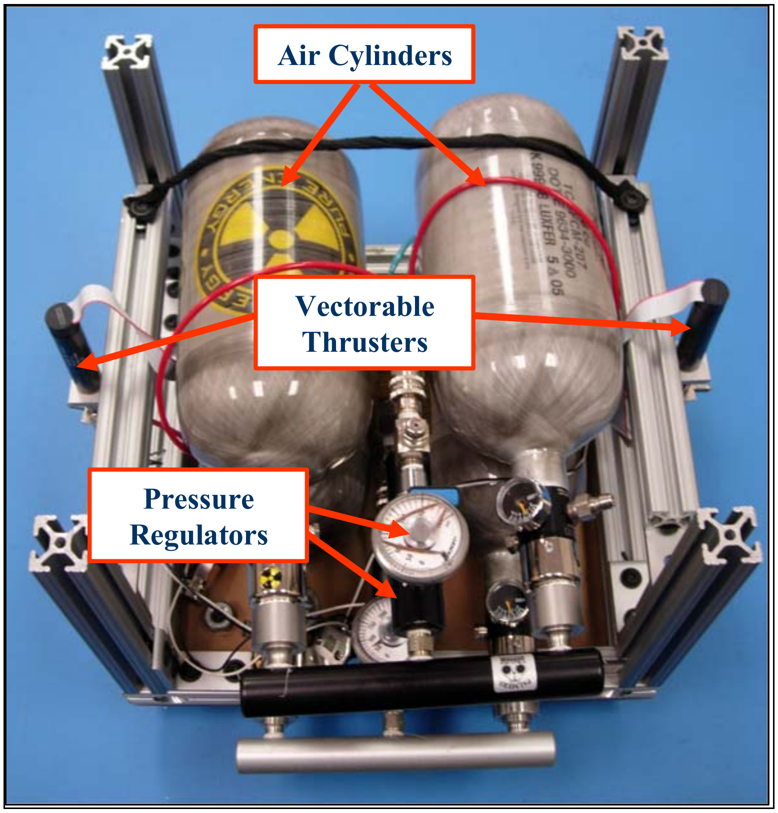


Figure 7. Lower Module Layout

The key enhancement to the air supply system involves the separation of the floatation and propulsion air subsystems, shown in Figure 8. Each subsystem is now supplied by two 68 cubic inches, 3000 PSI carbon fiber air cylinders, for a total of four cylinders, doubling the onboard air storage capacity. This arrangement has the benefit of allowing individual testing and actuation of the subsystems. This arrangement also ensures that the floatation subsystem receives an adequate air supply regardless of the demand placed on the air supply system by the thrusters. Another improvement was the additional of a fourth air pad and relocating the four air pads in four corners of the bottom of the lower module. This fourth air pad ensures that the simulator will float easily with the added weight of the additions made to the simulator. The floatation system operates at 40 PSI while the propulsion system operates at 120 PSI. Further details of the air system along with filling the air cylinders are described in Ref. [8].

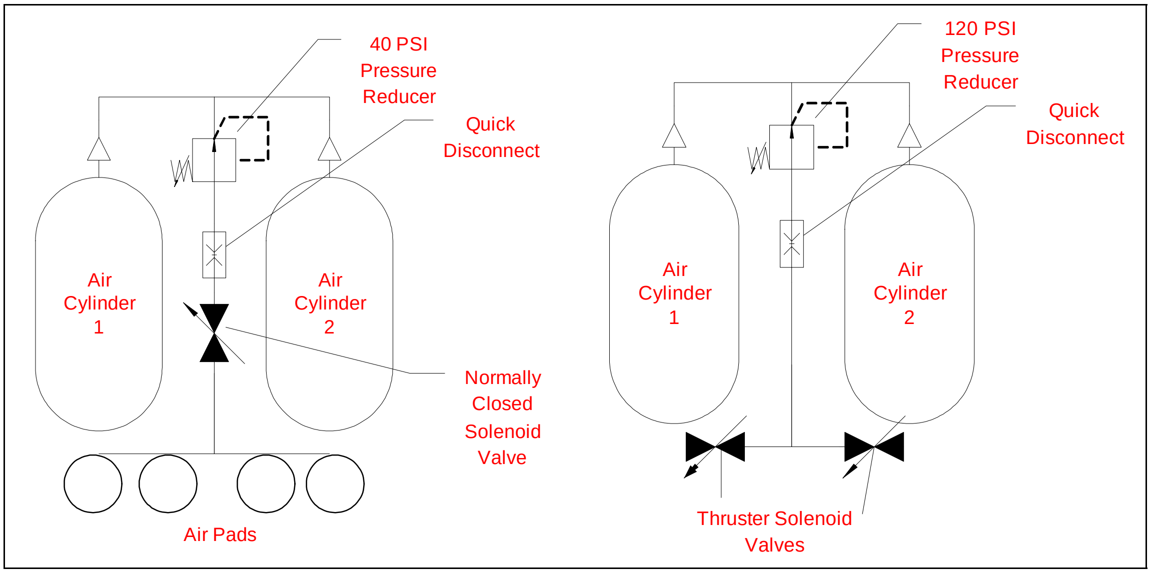


Figure 8. Air System Schematic (Ref. [8])

C. ELECTRICAL DISTRIBUTION SYSTEM IMPROVEMENTS

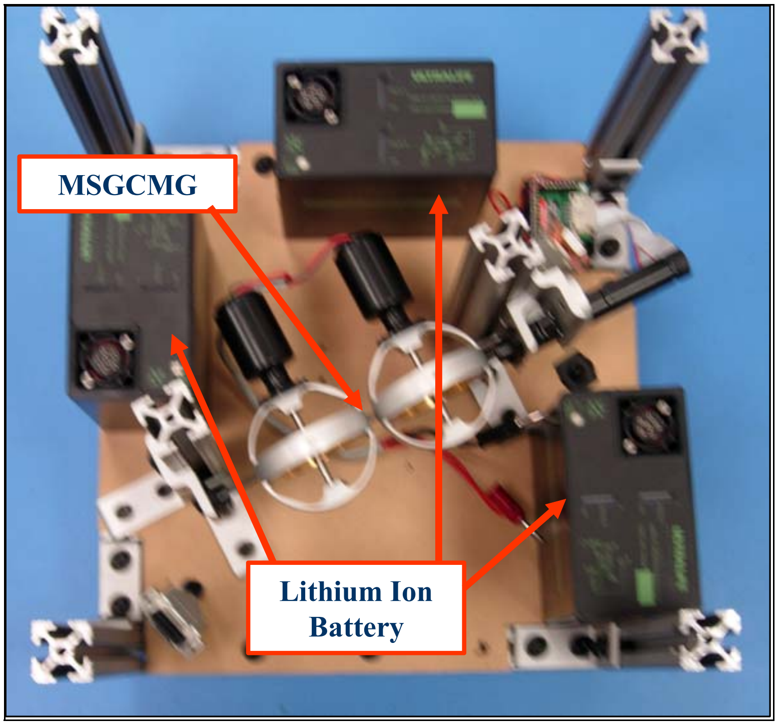


Figure 9. Middle Module Layout

They key enhancement to the electrical distribution system, shown in Figure 10, is the addition of another battery. This change, along with the addition of the two air cylinders for the air system, required a complete redesign of the electrical distribution system layout. The electrical system was moved from the lower module to the upper and middle modules, as shown in Figure 6.

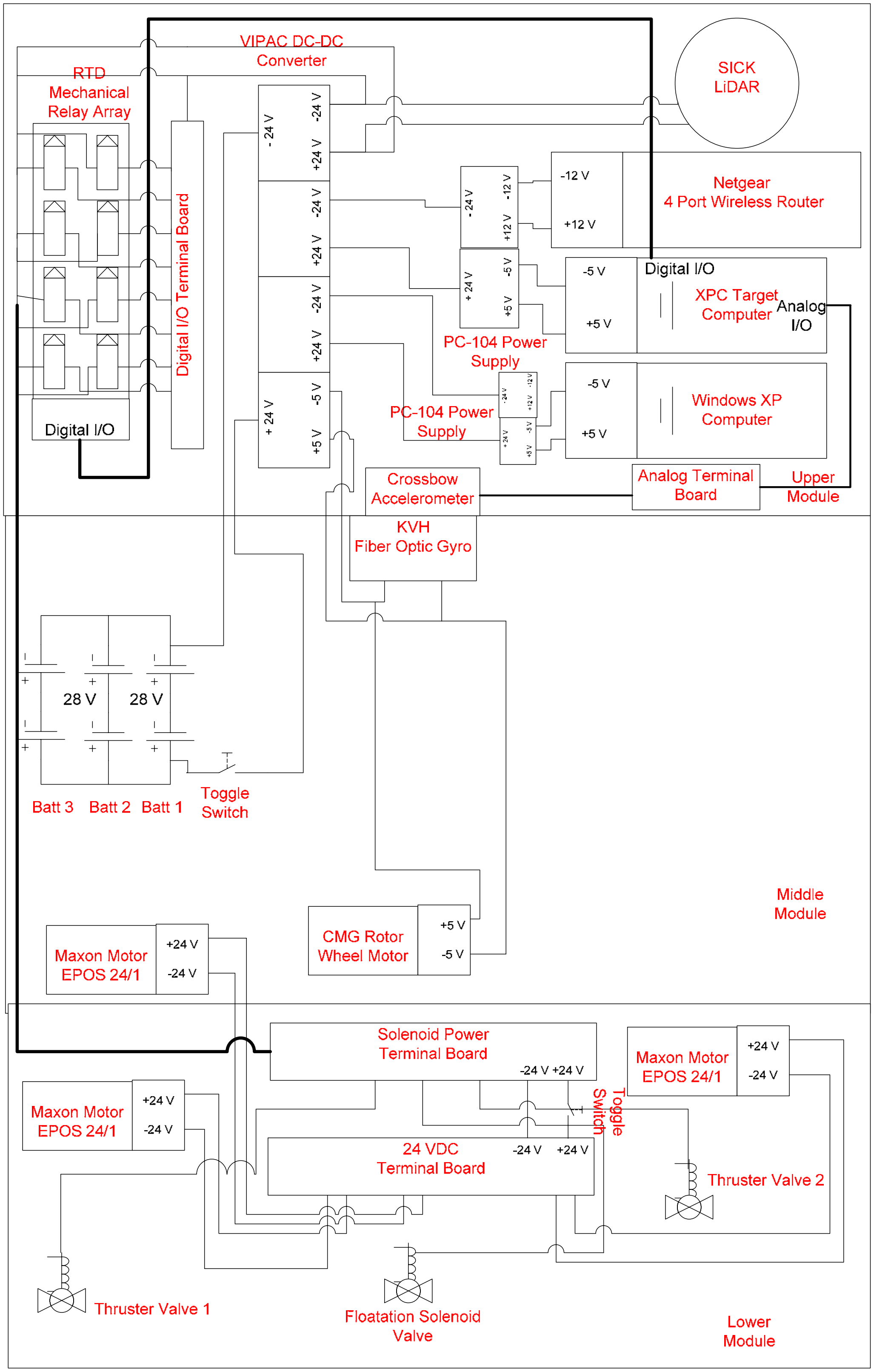


Figure 10. Electrical Distribution System Schematic (Ref. [8])

Three Ultralife UBBL02 Lithium Ion batteries are located in the middle module, surrounding the MSGCMG. The UBBL02 contains two internal cells, shown in Figure 11, each with a voltage of 16 VDC, that are wired in series by connecting the output pins five and three together and using pins two and four to connect the battery to the load. This configuration increases the output voltage to 32 VDC. Each battery has a six Ah capacity at 32 VDC, for a total capacity of 18 Ah for the spacecraft simulator.

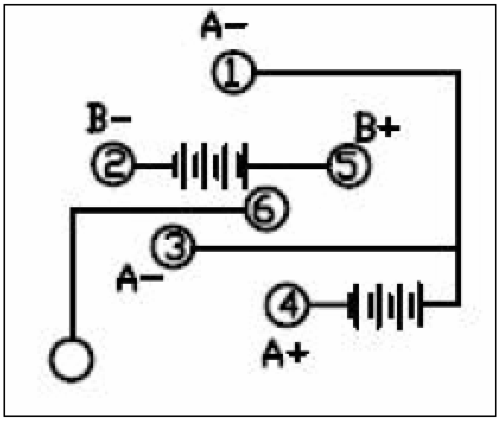


Figure 11. Ultralife UBBL02 Lithium Ion Battery Schematic (Ref. [10])

The batteries are connected thru a mechanical switch to a Vicor four output terminal VIPAC DC-DC converter array mounted in the upper module. The DC-DC converter array contains one five VDC DC-DC converter and three 24 VDC DC-DC converter. The DC-DC converter array itself draws approximately .25 A of current or approximately 8 W of power. Another key design change was redistributing the loads among the DC-DC converters, specifically the 24 VDC converters. Each PC-104 is connected to its own dedicated 24 VDC converter, with the third converter being used to power the LiDAR, the air solenoid valves, and the DC motors used to manipulate the MSGCMG and the vectorable thrusters. The power requirements for all of the individual components are listed in Table 2. The total power requirement the spacecraft simulator is approximately 100 W at 32 VDC.

Component	Voltage Reqs (Max Power)	Component	Voltage Reqs (Max Power)
(2) Versalogic EPM-CPU-10	5 [V] (21 [W])	SICK LD-OEM LiDAR	24 [V] (15 [W])
Netgear 4 Port Router	12 [V] (6 [W])	(3) ASCO Solenoid Valves	24 [V] (6 [W])
(3) Maxon Motor EPOS 24/1 Encoder	24 [V] (2 [W])	KVH DSP-3000 Fiber Optic Gyro	5 [V] (2 [W])
(3) Super Precision Gyro Motor	5 [V] (1 [W])	Crossbow CXL02TG3 Accelerometer	5 [V] (10 [mW])
VICAP DC-DC Converter	24-40 [V] (8 [W])	Metris iGPS	Own Power Source

Table 2. Spacecraft Simulator Electrical Loads (Ref. [8])

D. COMMAND AND DATA HANDLING SYSTEM

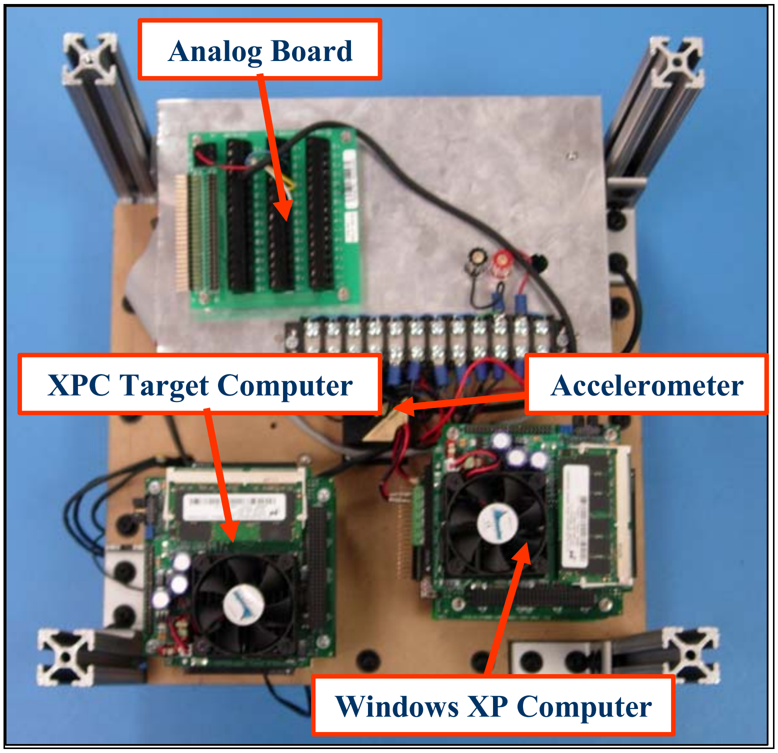


Figure 12. Upper Module Layout

The Command and Data Handling (C&DH) system is responsible for all of the control functions and communications for the spacecraft simulator. The C&DH consists of two PC-104 computers, a mechanical relay board, a 50 terminal connection board, and a wireless router. All of these components, with the exception of the wireless router, are located in the upper module. The wireless router is located underneath the sensor deck. Due to current hardware limitations, there is also an off board computer running Lynx that is used to retrieve Indoor Global Positioning System (iGPS) information.

One PC-104 is used to run the MATLAB XPC Target Operating System delivering real-time control capability to the simulator while the second PC-104 is used to run Windows XP and processes the raw Laser Scanner and iGPS information into useable data and then sends this processed data to the control computer. The size, capability and rapid upgradeability of embedded PC-104 computers drove their selection over other types of CPUs. Both PC's are equipped with Ethernet and RS-232 serial ports used to connect them to each other through the router and to the pertinent sensors respectively. (Ref. [8])

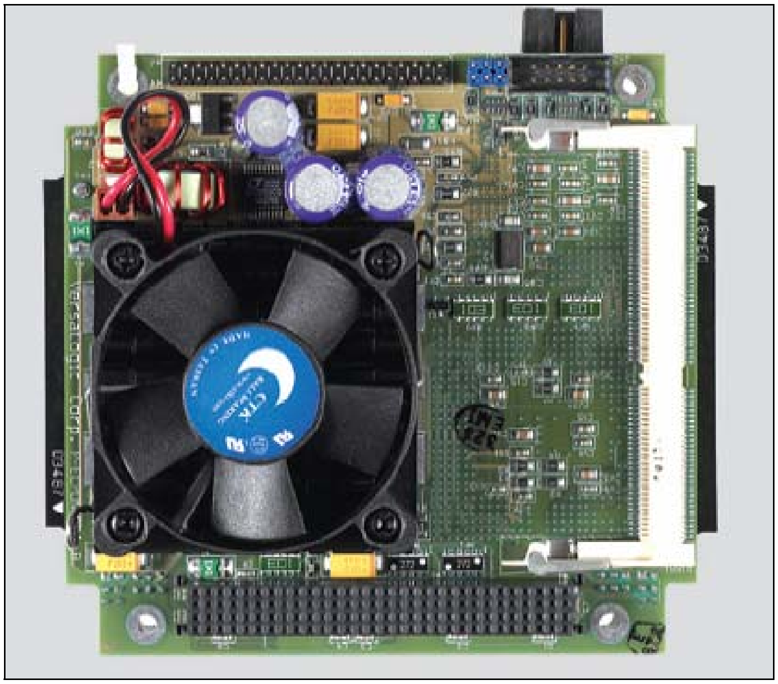


Figure 13. Versalogic Jaguar EPM-CPU-10 PC-104 (Ref. [11])

The PC-104's used for the spacecraft simulator are the Versalogic Jaguar EPM-CPU-10 PC-104, shown in Figure 13. The Versalogic Jaguar is running on an 850 MHz

Pentium III processor with 256 MB DRAM (Dynamic Random Access Memory). It is equipped with KVM, dual USB, Ethernet, dual RS-232 communication ports, and IDE disk drive and a 3.5 inch floppy drive connections. (Ref. [8])

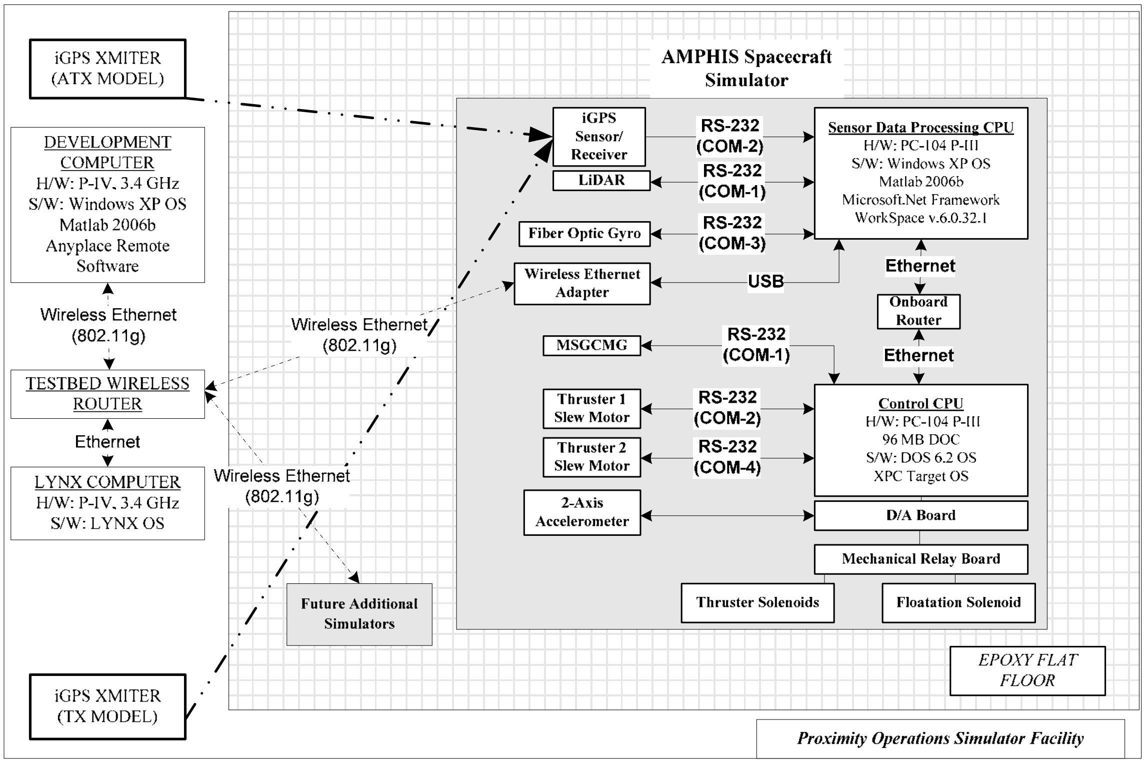
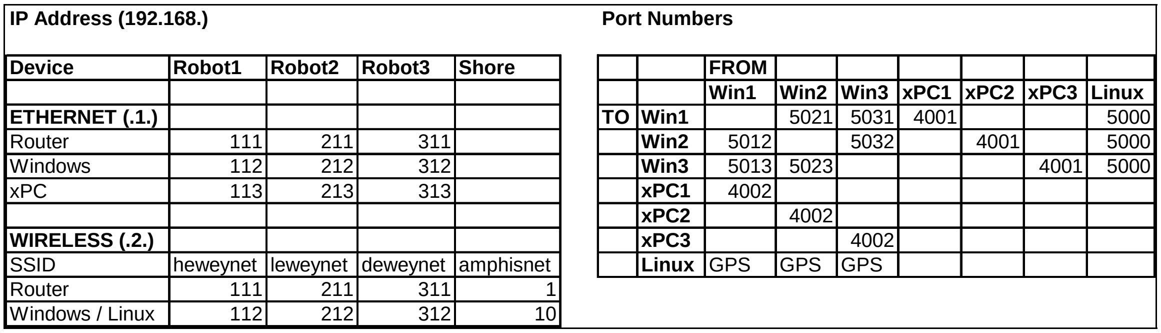


Figure 14. AMPHIS Data Handling Schematic

The overall data processing scheme for the AMPHIS test bed is depicted in Figure 14. The communication between the two onboard computers of the spacecraft simulator is handled thru the onboard wireless router via wired Ethernet connections. All communication off the spacecraft simulator is thru the Windows XP computer and its wireless Ethernet adapter. A wireless router for the AMPHIS test bed links the various spacecraft simulators, the test bed development computer, and the Lynx computer together. The IP address and port configuration scheme for the test bed facility, including the individual spacecraft simulators, is depicted in Table 3. Further discussion of the TCP/IP architecture is provided in Ref. [9].



IP Address (192.168.)

Device	Robot1	Robot2	Robot3	Shore
ETHERNET (.1.)				
Router	111	211	311	
Windows	112	212	312	
xPC	113	213	313	
WIRELESS (.2.)				
SSID	heweynet	leweynet	deweynet	amphisnet
Router	111	211	311	1
Windows / Linux	112	212	312	10

Port Numbers

		FROM						
		Win1	**Win2**	**Win3**	**xPC1**	**xPC2**	**xPC3**	**Linux**
TO	**Win1**		5021	5031	4001			5000
	Win2	5012		5032		4001		5000
	Win3	5013	5023				4001	5000
	xPC1	4002						
	xPC2		4002					
	xPC3			4002				
	Linux	GPS	GPS	GPS				

Table 3. IP Address and Port Configuration Scheme (Ref. [9])

1. XPC Target Computer

The PC-104 stack used for the XPC Target computer contains, in addition to the Jaguar computer, a TRI-M Engineering 75W PC-104 power supply, a M-Systems 96MB Disk On Chip (DOC) flashdisk, a Versalogic Quad RS-232 Module, and a Diamond Systems DMM-32X-AT 16-bit Analog I/O Module. The XPC Target computer is connected via RS-232 connections to the MSGCMG, the two vectorable thrusters' slewing motor, and may be connected to the Fiber Optic Gyro. The computer is connected to the onboard wireless router via its own Ethernet wire connection. The 12 VDC output of PC-104 power supply is to power the onboard wireless router. Detailed instructions for the setup of the XPC Target computer are in Appendix B. The SIMULINK model used for XPC Target control computer is shown in Figure 15. Full detail on the XPC Target SIMULINK model development is provided in Ref. [9].

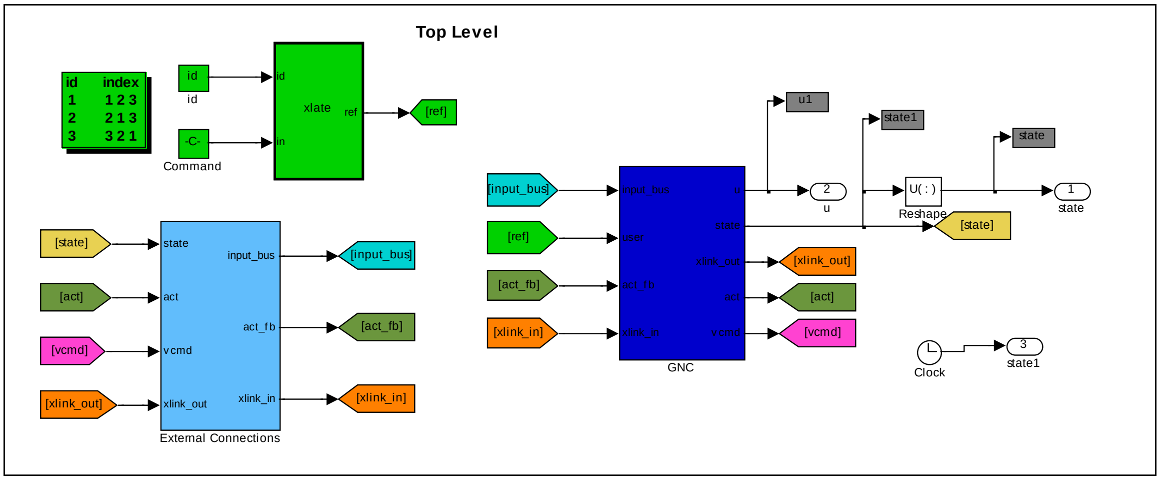


Figure 15. XPC Target Computer SIMULINK Model (Ref. [9])

A 50 pin ribbon connecter is used to connect the analog interface of the DMM-32X-AT module to a 50 terminal connection board. This connection board is used to connect the accelerometer to the computer. A 34 pin ribbon cable is used to connect to the digital interface of the DMM-32X-AT module to a RTD DMR-8 Mechanical Relay Output Board, both shown in Figure 16. The DMR-8 has a 50 pin connection and uses a different wiring scheme than the DMM-32X-AT. Therefore, the opposite end of the 34-pin connection is changed with a 50 pin female connecter. Only the first 8 bits (A0-A7, pins 1-8) of the DMM-32X-AT are used to communicate with the DMR-8 along with the five VDC and digital ground connection (pin 33 and 34). Wires one thru eight of the 34 pin ribbon cable are connected onto connections 33, 35, 37, 39, 41, 43, 45, and 47 respectively, corresponding to DIN zero thru seven on the DMR-8. Wires 33 (5 VDC) and 34 (Dgnd) are connected to connections 49 and 50 respectively on the DMR-8 to provide power to DMR-8.

A7	1	2	A6
A5	3	4	A4
A3	5	6	A2
A1	7	8	A0
B7	9	10	B6
B5	11	12	B4
B3	13	14	B2
B1	15	16	B0
C7	17	18	C6
C5	19	20	C4
C3	21	22	C2
C1	23	24	C0
Latch	25	26	Ack
NC	27	28	NC
(RS-232) Tx	29	30	Rx (RS-232)
(RS-485) Rx/Tx +	31	32	Rx/Tx - (RS-485)
+5V	33	34	Dgnd

DIN8	1	2	EXTINT
DIN9	3	4	DIGITAL GND
DIN10	5	6	DIGITAL GND
DIN11	7	8	DIGITAL GND
DIN20	9	10	DIGITAL GND
DIN21	11	12	DIGITAL GND
DIN22	13	14	DIGITAL GND
DIN23	15	16	DIGITAL GND
DIN12	17	18	DIGITAL GND
DIN13	19	20	DIGITAL GND
DIN14	21	22	DIGITAL GND
DIN15	23	24	DIGITAL GND
DIN16	25	26	DIGITAL GND
DIN17	27	28	DIGITAL GND
DIN18	29	30	DIGITAL GND
DIN19	31	32	DIGITAL GND
DIN0	33	34	DIGITAL GND
DIN1	35	36	DIGITAL GND
DIN2	37	38	DIGITAL GND
DIN3	39	40	DIGITAL GND
DIN4	41	42	DIGITAL GND
DIN5	43	44	DIGITAL GND
DIN6	45	46	DIGITAL GND
DIN7	47	48	DIGITAL GND
+5 VOLTS	49	50	DIGITAL GND

Figure 16. DM-32X-AT Digital I/O Connection Diagram (Ref. [12]) and DMR-8 Mechanical Relay Board Connection Diagram (Ref. [13])

2. DMR-8 Mechanical Relay Board

The DMR-8 isolates the low power TTL signal of the digital outputs of the XPC Target computer from the high voltage (24 VDC) power supply used to actuate the solenoids for the vectorable thrusters, the air supply to air pads of the floatation system, and future uses such as actuating docking mechanisms. Each mechanical relay has a normally open (NO), a normally closed (NC), and a common connection (COM), as shown in Figure 17. When a high TTL signal is sent to the mechanical relay, the relay closes the NO terminal, opens the NC terminal, and illuminates the LED.

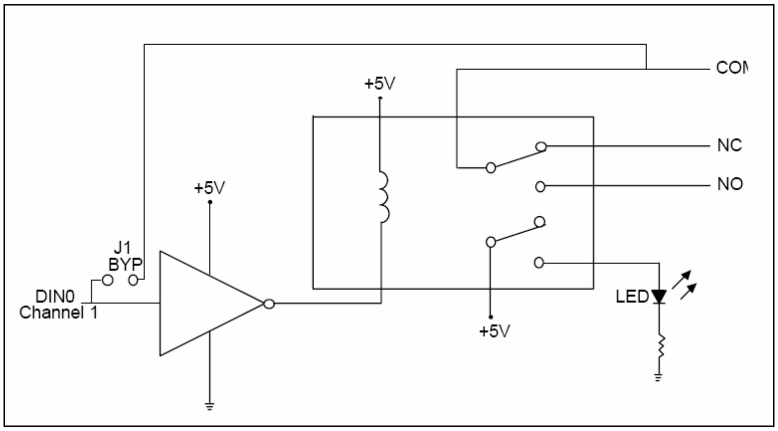


Figure 17. Mechanical Relay Circuit Diagram (Ref. [13])

All of the COM connections on the DMR-8 are connected thru a terminal board to a -24 VDC connections of one of the 24 VDC regulators on the DC-DC converter. This allows the mechanical relay board to interrupt the ground connection on attached devices, effectively turning them on or off. The NS or NO connections are combined together with a +24 VDC and ground to a combined wire bundle with a 15 pin serial connector on the end, according to the wiring scheme depicted in Table 4. The 15 pin serial connector allows for easy disconnection in order to separate the individual modules of the spacecraft simulator. The opposite end of the combined wire bundle is connected to a screw terminal board, set up in pairs of +24 VDC and the individual relay board connections to allow for easy connection of individual devices.

Pin Number	Color	Connection
1	Brown	DIN 0 (NS)
2	Purple	DIN 1 (NO)
3	Orange	DIN 2 (NO)
4	Yellow	DIN 3 (NO)
5	Light Blue	DIN 4 (NO)
6	White	DIN 5 (NO)
7	Gray	DIN 6 (NO)
8	Green	DIN 7 (NO)
9	Red	+24 VDC
15	Black	GRND

Table 4. 15 Pin Serial Connector Wiring Scheme

3. Windows XP Computer

The PC-104 stack used for the Windows XP computer contains, in addition to the Jaguar computer, a RTD 75W PC-104 power supply, a Fujitsu 3.5” laptop hard drive, and a Versalogic Quad RS-232 Module. The Windows XP computer is connected via RS-232 connections to the iGPS and LiDAR sensors, and may be connected to the Fiber Optic Gyro. The computer’s USB connection is used to connect it to a wireless Ethernet adapter and the computer is connected to the onboard wireless router via its own Ethernet wire connection. The PC-104 five VDC output of the PC-104 power supply is used to power the hard drive. The Windows XP computer runs the host software for the iGPS as well as SIMULINK. Detailed instructions for the setup of the Windows XP computer are found in Appendix C. It is this SIMULINK model, shown in Figure 18, that runs all of the communications for the spacecraft simulator as well as do all of the LiDAR processing. The communication architecture and LiDAR processing are discussed in detail in Ref. [9].

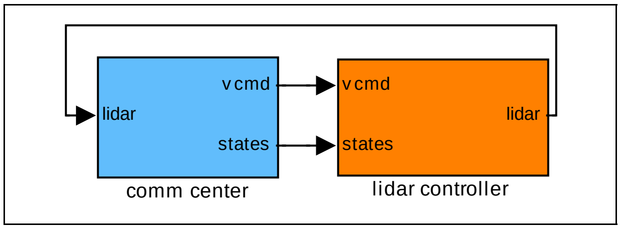


Figure 18. Windows XP Computer SIMULINK Model

E. SENSORS

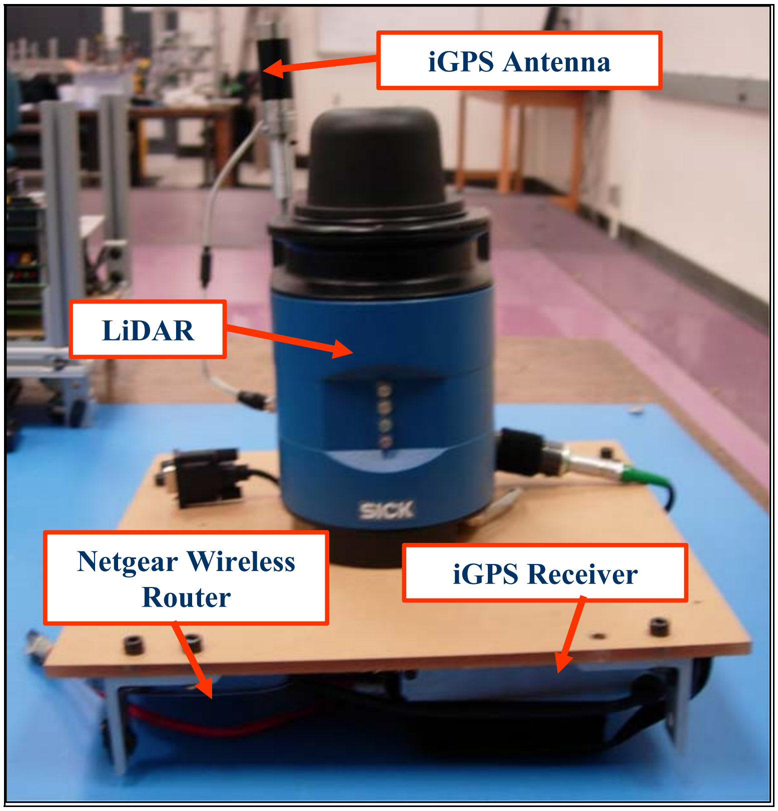


Figure 19. Sensor Module Layout

1. Fiber Optic Gyro

Figure 20. KVH DSP-3000 Fiber Optic Gyro (Ref. [14])

A KVH fiber optic gyro (FOG) provides the attitude measurement requirements for the spacecraft simulator. The KVH DSP-3000 FOG selection is based on its compact, lightweight frame, its low rate bias term and its serial interface. The Digital 100Hz Asynchronous (KVH part number 02-1222-01) variant is used. The KVH FOG specifications are provided in Table 5.

Attribute	**Rating**
Performance	
Maximum Input Rate	$\pm 375^\circ/\text{sec}$
Update Rate	100 Hz
Angle Random Walk (noise)	$4^\circ/\text{hr}/\sqrt{\text{Hz}}$
Initialization Time	< 5 sec
Bias	
Offset (room temp)	$\pm 20^\circ/\text{hr}$
Stability (room temp)	$1^\circ/\text{hr},\ 1\sigma$
Temperature Sensitivity ($<1^\circ\text{C}/\text{min}$)	$6^\circ/\text{hr},\ 1\sigma$
Electrical	
Input Voltage	+5 VDC $\pm 10\%$
Power Consumption	3 watts max, 2 watts norm
Output	
Baud Rate	38,400 Baud (RS-232)
Parity	None
Data bits	8
Stop bits	1
Flow Control	None
Physical	
Dimensions	88.9 mm x 58.42 mm x 33.02 mm
Weight	0.27 kg

Table 5. KVH DSP-3000 Specifications (Ref. [14])

As shown in 0, the gyro senses rotation on an axis perpendicular to the plane of the base plate about its centroid. The gyro is mounted underneath the upper module of the spacecraft simulator. This ensures that the gyro is perpendicular to the floor of the laboratory with its centroid located approximately centerline to the spacecraft simulator, aligning the gyro's axis of rotation with the Z-axis of the spacecraft simulator. Looking at the gyro overhead, a clockwise rotation corresponds to a positive output. The inverted mounting underneath the upper platform corrects this convection to coincide with the normal right-hand rule convention with the X and Y axes of the laboratory.

The DSP-3000 is equipped with a 15-pin interface connector, shown in Figure 21, located on one of the long ends of the gyro. With the connector positioned at the top of the gyro wall, pin 1 is located on the right side when viewing the connector head-on. Tyco Electronics' single-row Dualobe connector (Tyco part number SSM015L2HN, KVH part number 32-0780) is used to interface with the gyro connector. This mating connector provides 12 inch leads.

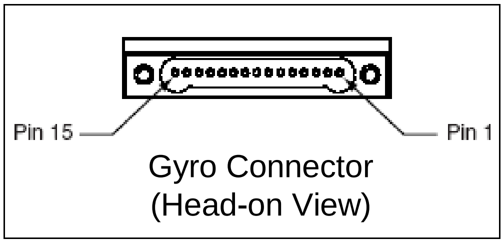


Figure 21. KVH DSP-3000 Interface Connector (Ref. [14])

Figure 22 illustrates the wiring diagram for the gyro. Pin one (power) is connected to the positive five VDC output of the DC-DC converter, while pins two and three (power and case ground) are connected to the DC-DC converter negative five VDC output. Pins nine, 10, and 11 correspond to the RS-232 communication for the gyro. Pin nine (Tx), 10 (Rx), and 11 (signal ground) are combined to form the pins two, three, and five of a RS-232 female connector respectively. This RS-232 connecter can be connected to one of the serial COM ports on either the Windows XP or XPC Target PC-104's.

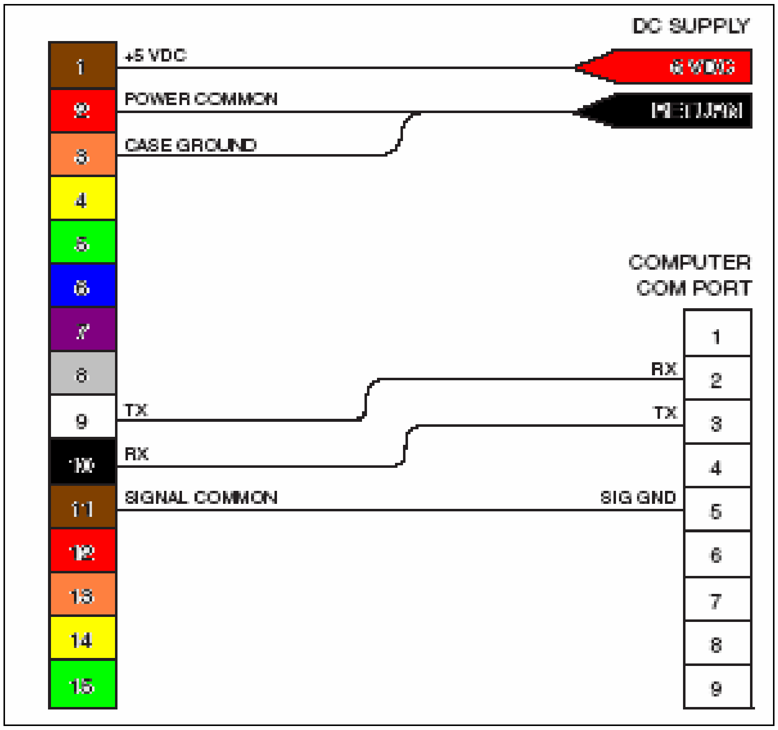
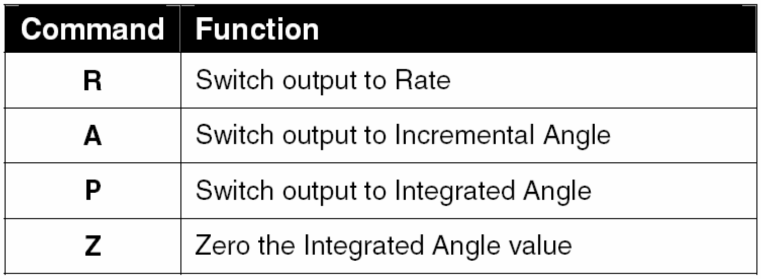


Figure 22. KVH DSP-3000 Wiring Diagram (Ref. [14])

There are several data output formats that the KVH DSP-3000 gyro can provide. They include angular rate, incremental angle, and integrated angle. The output mode is determined by sending a single ASCII character to the gyro without a carriage return or line feed. An additional command is available to zero the integrated angle value. A list of commands is listed in Table 6. A command may need to be sent more than once for the command to execute.



Command	Function
R	Switch output to Rate
A	Switch output to Incremental Angle
P	Switch output to Integrated Angle
Z	Zero the Integrated Angle value

Table 6. KVH DSP-3000 User Commands (Ref. [14])

The output data is in the form of a continuous ASCII text stream consisting of two decimal words separated by three space characters and followed by a carriage return/line feed sequence. The first data word begins four space characters after the line feed character. Its content represents angular rate (degs/sec), incremental angle (deg), or integrated angle (deg) to six decimal point precision. The second data word consists of a single ASCII character of either zero for fault or one for data valid. For the spacecraft simulator, only the rate mode is used. Angular position is found by integrating the rate information.

Two separate methods were developed to retrieve this information from the gyro, allowing the gyro sensor to be connected to either the XPC Target computer or the Windows XP computer running SIMULINK. Figure 23 and Figure 24 illustrate the SIMULINK models used. Appendix E contains the MATLAB code used for the gyro.

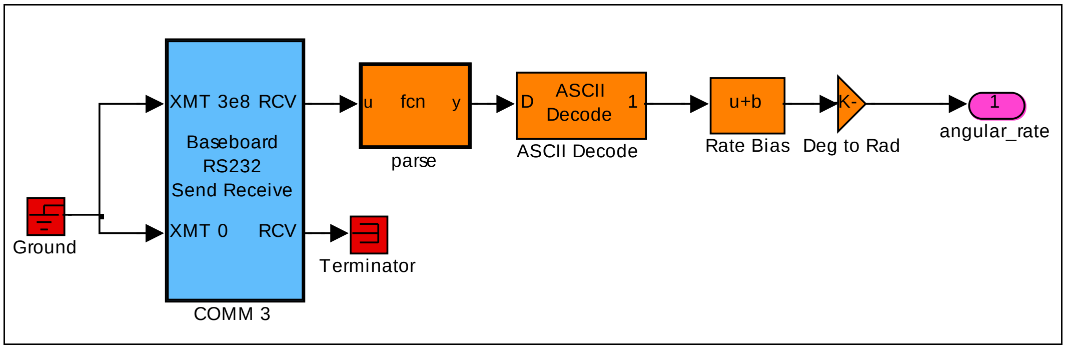


Figure 23. XPC Target Gyro SIMULINK Model

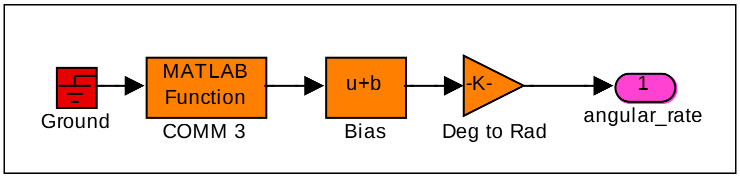


Figure 24. Windows XP Gyro SIMULINK Model

A rate bias term must be subtracted out from the measured rate in order to account for the rotation of the earth. The angular rate due to the earth's rotation can be found using equation(1).

$$\text{Earth Rate (deg/hour)} = -15.04107 * \sin(latitude) \qquad (1) \text{ (Ref. [14])}$$

2. Accelerometer

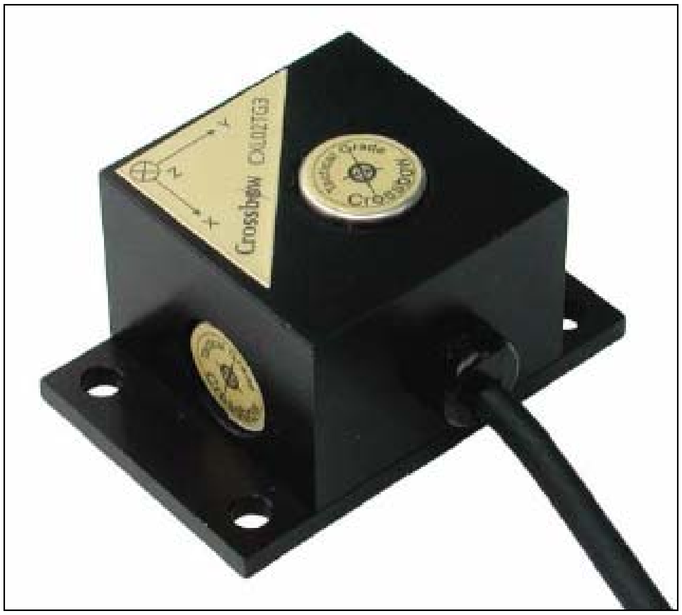


Figure 25. Crossbow CXL02TG3 Tri-Axial Precision Accelerometer (Ref. [15])

A Crossbow CXL02TG3 Tri-Axial Precision Accelerometer provides inertial measurements for the spacecraft simulator for the two axes of translational motion. The accelerometer, shown in Figure 25, is mounted in the upper module, centerline, with two of its axes (X and Y) aligned the X and Y body axes of the space craft simulator respectively. The third axis (Z-axis) is not used, since the spacecraft simulator has no movement along that axis. Table 7 lists the Crossbow accelerometer specifications.

Attribute	**Rating**
Performance	
Input Range	±2 g
Bias Stability	±8.5 *m*g
Sensitivity	833 *m*V/g
Noise at 100 Hz Bandwidth	.6 *m*g rms
Electrical	
Input Voltage	+5 VDC
Power Consumption	1.5 mW
Output	
Zero-g Voltage	2.49-2.51 VDC
Physical	
Dimensions	2.81 mm x 5.68 mm x 3.65 mm
Weight	0.11 kg

Table 7. Crossbow CXL02TG3 Specifications (Ref. [15])

The Crossbow accelerometer is equipped with an analog interface allowing for direct connection through the Diamond Systems DMM-32X-AT 16-bit Analog I/O PC-104 module to XPC Target computer. Table 8 depicts the wiring scheme for the accelerometer. The five VDC and ground wires are connected to terminals 49 and 50 respectively on the 50 pin analog terminal board. This enables the accelerometer to be powered by the XPC Target computer. The X-axis and Y-axis wires are connected to terminals 25 and 27 respectively. The Z-axis wire is not used due to no movement in the Z direction and the temperature wire is not used due to the near constant temperature of the laboratory environment.

Color	Function
Red	5 VDC
Black	GRND
White	X-axis
Yellow	Y-axis
Green	Z-axis
Blue	Temperature

Table 8. Crossbow Accelerometer Wiring Scheme (Ref. [15])

The output from the X and Y axis leads is in the form of an analog signal ranging from zero to five volts. The calibration data provided by Crossbow relates output voltage to G's of acceleration. Zero-G voltage is approximately 2.52 volts. The sensitivity of the output voltage is approximately $.845\, volts / G$. Figure 26 illustrates the SIMULINK model used in XPC Target to communicate with the accelerometer.

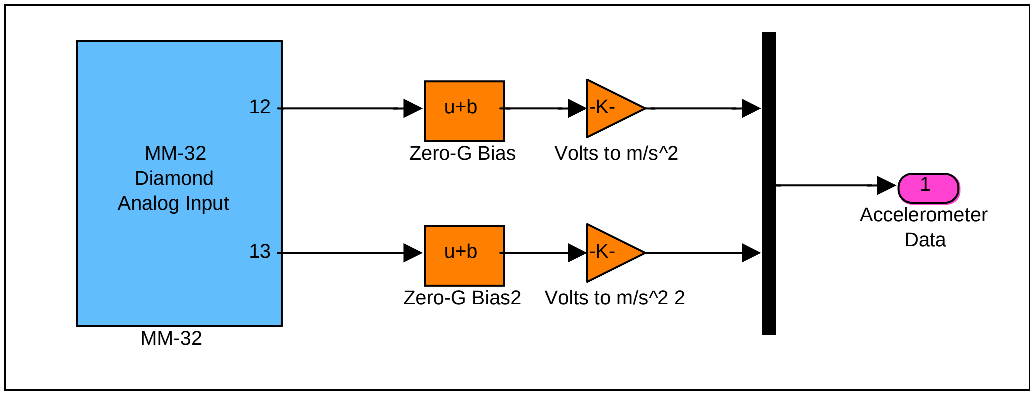


Figure 26. Accelerometer SIMULINK Model

3. Indoor Global Positioning System (iGPS)

The spacecraft simulator and POSF are equipped with the Indoor Global Positioning System (iGPS) by Metris. This system is similar to the on-orbit GPS in that it is capable of providing high precision position determination. However, unlike the use of radio signals by orbiting GPS satellites, the iGPS uses laser transmitters to cover the test bed with infrared light that is detected and then processed using on-board software, WORKSPACE. (Ref. [16]) The WORKSPACE software is loaded and run on the onboard the Windows XP computer. The procedures for setting up iGPS and installing the WORKSPACE software for use with the AMPHIS test bed are found in Ref. [6] and Ref. [8].

The iGPS receiver and battery pack are fixed underneath the top sensor deck. The iGPS antenna is located on top of the LiDAR sensor to ensure that it has continuous line of sight with the iGPS transmitters located in the POSF. This arrangement causes a small ‘blind spot’ for the LiDAR sensor, but the software developed in Ref. [9] is take it into account. The iGPS is connected via RS-232 cable to the one of the COM ports on the onboard Windows XP computer.

Due to temporary limitations with the current WORKSPACE software, an off board computer running the Lynx operating system, uses C code to retrieve the iGPS information via IP/TCP communication and relays that information back to the onboard Windows XP computer via UDP communication, which then passes the information onto the XPC Target computer for state estimation calculations. This awkward arrangement will be resolved with future versions of iGPS.

4. Light Detection and Ranging (LiDAR) Sensor

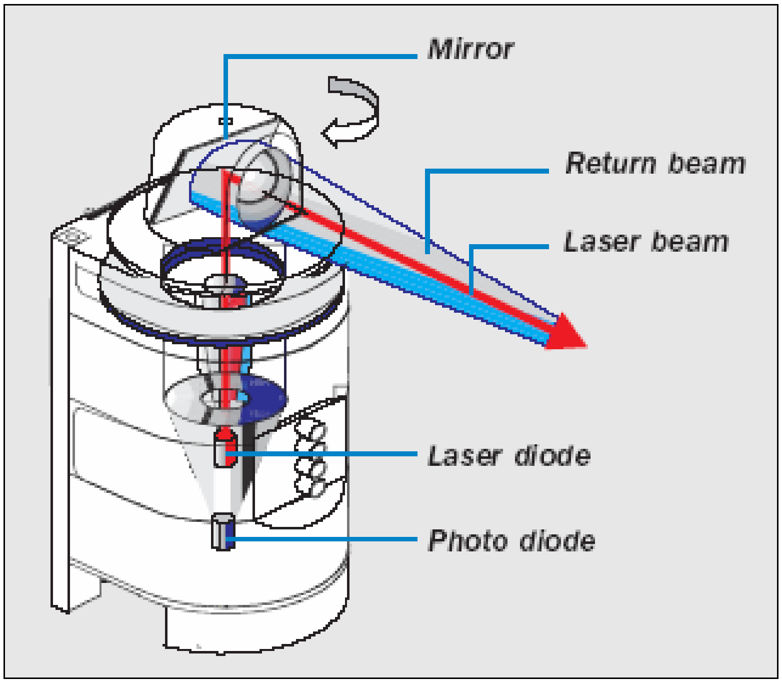


Figure 27. SICK LD-OEM Laser Scanner (Ref. [20])

The primary sensor for determining the relative bearing and range of other spacecraft simulators on the AMPHIS test bed is the SICK LD-OEM Laser Scanner, shown in Figure 27. The mirror assembly on the top of the scanner rotates about the Z axis of the simulator, providing range and bearing information from the return of the reflected transmitted laser pulses in the X-Y plane, providing the spacecraft simulator at 2-D view of the AMPHIS test bed environment. The LiDAR provides the spacecraft simulator with a 360 degree view of the POSF, as shown in Figure 28.

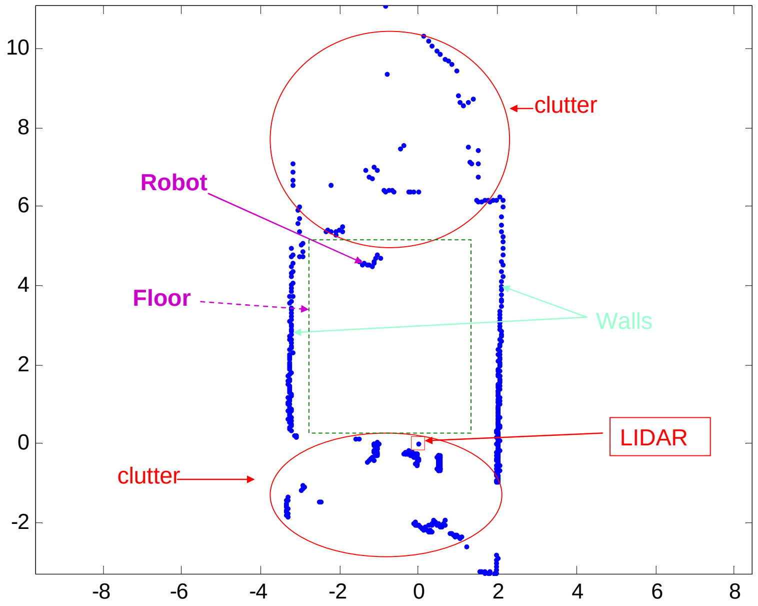


Figure 28. Sample LiDAR Image of POSF (Ref. [9])

The LiDAR sensor is mounted on top of the sensor deck of the spacecraft simulator. It is powered with 24 VDC from the one of the 24 VDC regulators on the DC-DC converter. The LiDAR is connected via a RS-232 connection to the onboard Windows XP computer. The software developed to communicate with the LiDAR is run as part of the Windows XP computer SIMULINK Model along with the communication blocks. The software development for the LiDAR is discussed in detail in Ref. [9].

F. ACTUATORS

1. Miniature Single Gimbaled Control Moment Gyro (MSGCMG)

A MSGCMG assembly mounted in the second module, shown in Figure 9, provides attitude control for the spacecraft simulator. A Super Motorized Precision Gyroscope by Educational Innovations, Inc. is at the core of the assembly. The gyroscope is rotated on the plane of the spinning rotor wheel by a graphite brush DC motor, commanded through an encoder both manufactured by Maxon Motor. A RS-232 interface provides the hardware interface between the XPC Target control computer and a Maxon Motor EPOS 24/1 digital motion controller. A second gyroscope was added to give additional attitude control authority. The hardware integration and development of the XPC Target control software for the EPOS 24/1 will be discussed in a proceeding section. The technical characteristics for the MSGCMG are found in Table 9. Further discussion about the development of the MSGCMG and its performance can be found in Ref. [8].

Rotor Wheel	
Moment of Inertia Wheel $\left(J_w\right)$	$3.717x10^{-3}$ [kg-m^2]
Maximum Momentum Wheel $\left(h_w\right)$	$49.4x10^{-3}$ [N-m-s]
Maximum Wheel Speed	±1256.6 [rad/s] (12000 rpm)
Power Supply	5 [VDC]
Gimbal Motor & Encoder	
Maximum Gimbal Rate	6.95 [rad/s] (398 deg/s)
Maximum Gimbal Acceleration	109.9 rad/s^2 (6297 deg/s^2)
Maximum Gimbal Torque	$4.06x10^{3}$ [N-m]
CMG	
Total Mass of CMG (including mounting hardware)	1.148 [kg]
Power Supply	9-24 VDC
Maximum Output Torque	.344 [N-m]
Power	< 24 [W]
Interface	RS-232
Dimensions	.02 [m^2]

Table 9. MSGCMG Technical Specifications (Ref. [8])

2. Vectorable Thruster

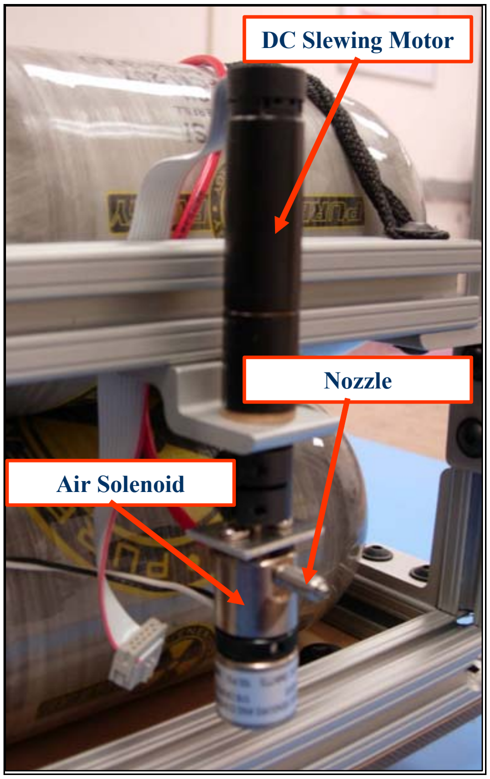


Figure 29. Vectorable Thruster

A pair of independently controlled vectorable thrusters, shown in Figure 29, provide translational as well as attitude control for the spacecraft simulator. The vectorable thrusters are mounted on opposite sides of the spacecraft simulator in the lower module, as shown in Figure 7. They are constructed of a Maxon brushless DC motor with a EPOS 24/1 digital motion controller used for slewing, a Predyne EH-2012 air solenoid used for thruster activation, and a Silvent MJ4 air nozzle to guide the air flow out of the solenoid. The hardware integration and development of the XPC Target control software for the EPOS 24/1 will be discussed in the proceeding section.

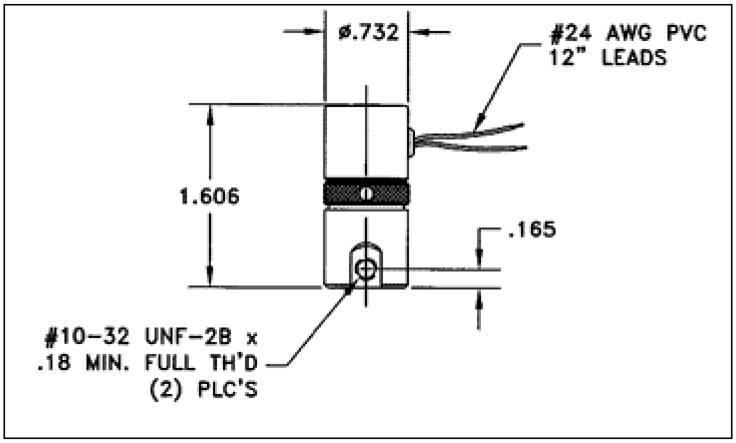


Figure 30. Predyne EH-2012 Air Solenoid Schematic (Ref. [17])

The EH-2012 air solenoid, shown in Figure 30, is a two-way normally closed electric valve. In order to actuate or open the solenoid valve, power (24 VDC) must be supplied to the valve. The 24 VDC power supply used to control the air solenoid is supplied thru the mechanical relay board, controlled by the XPC Target computer. A complete discussion on this hardware interface is discussed in the previous section on the C&DH system. The air solenoid characteristics are found in Table 10.

Parameter	**Value**
Max Pressure	100 psi
Orifice Diameter	1.6 mm
Cv Factor	.050
Minimum Cycle Time	3-5 milliseconds
Power Requirement	24 VDC (2 W)
Dimensions	40.8 mm x 18.6 mm (dia.)
Weight	.015 kg

Table 10. Predyne EH-2012 Air Solenoid Specifications (Ref. [17])

The digital output signal generated by the DMM-32X-AT Analog I/O module is controlled by the SIMULINK model shown in Figure 31. The two independent required thrust signals generated by the controller are sent to the MM-32 Diamond Digital Output SIMULINK control block. Relay blocks are used to ensure the required TTL value of .5 is sent to the digital output block when a thrust is required. The digital control block also controls the air solenoid used to actuate the floatation system. Time activated switches ensure that the thrusters and air pads are deactivated at the end of an experiment.

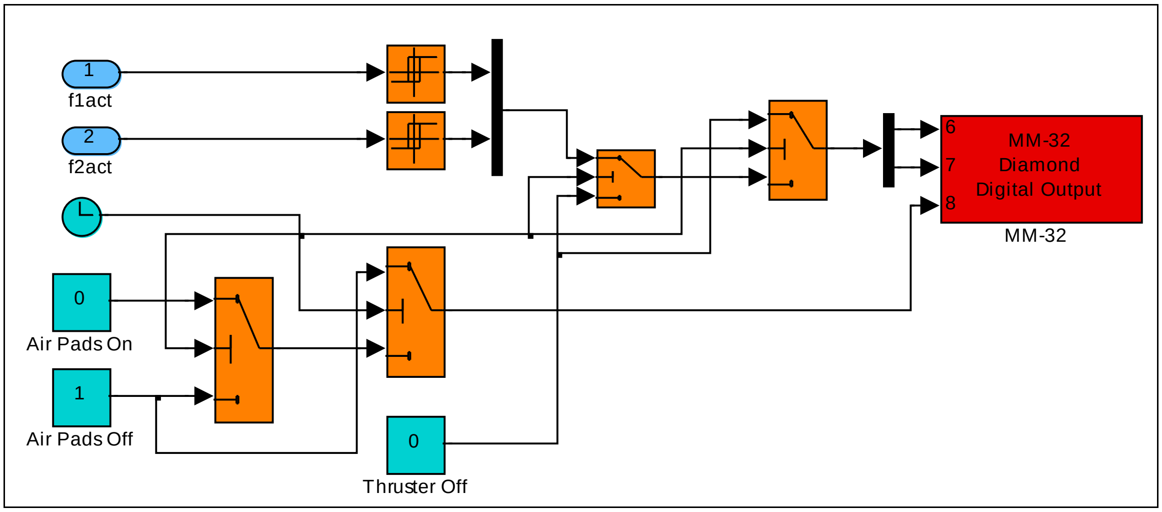


Figure 31. Air Solenoid Control SIMULINK Model

Calibration runs of continuous thrust, shown in Figure 32, were used to determine the actual thrust output of the thrusters. The iGPS sensor was used to record position information. A first order transfer function, shown in equation(2), was used to calculate the first and second derivatives, corresponding to velocity and acceleration, from the position information.

$$\frac{s}{s+1} \tag{2}$$

From the data in Figure 32, a near constant acceleration of approximately.0075 m/s^2 . Based on the spacecraft simulator mass of 37 kg, this constant acceleration value corresponds to a constant thrust value of .28 N for each thruster. Based on the physical location of the vectorable thrusters, they each have a moment arm of .15 meters. This allows the thrusters to apply a max torque on the spacecraft simulator of .084 Nm, compared to the .344 Nm available from MSGCMG. However, the angular momentum output of the MSGCMG is limited to .049 Nms, where as the angular momentum supplied by the vectorable thrusters is essentially limitless, only limit being air supply.

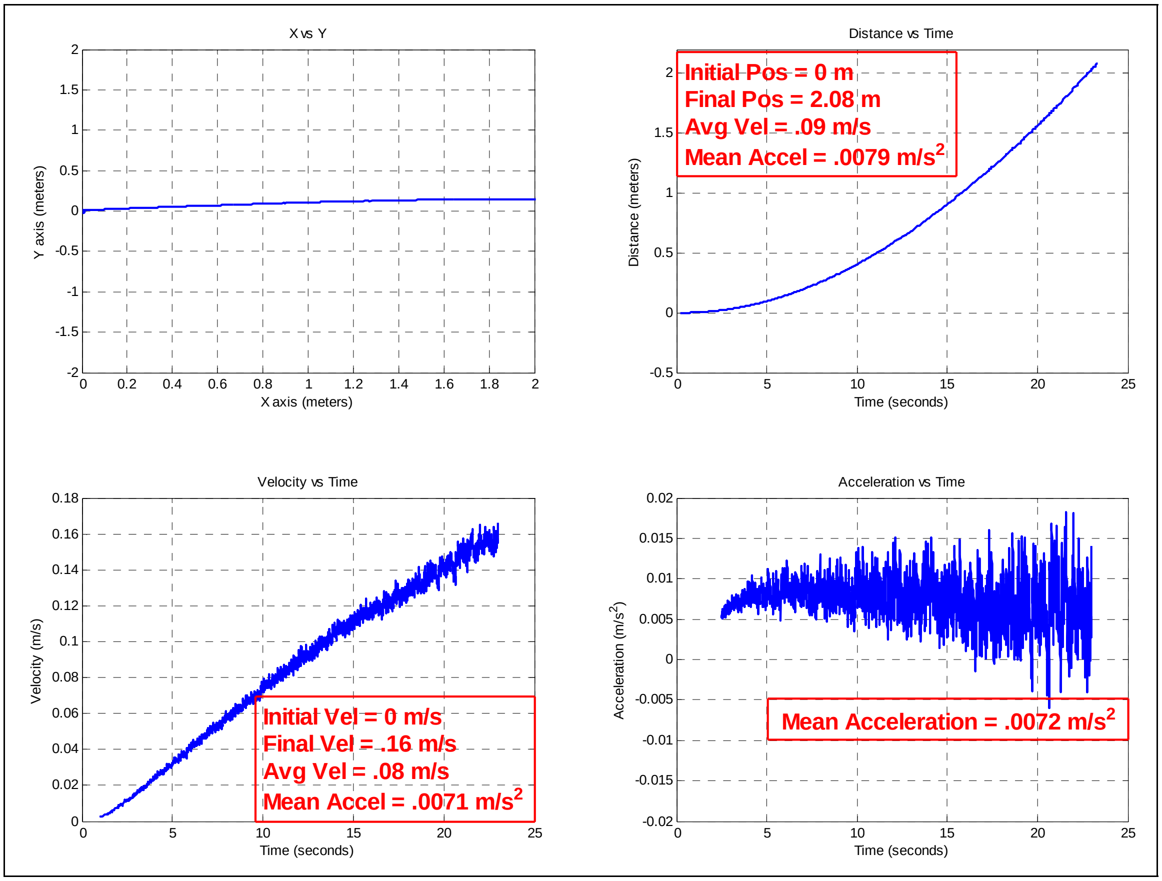


Figure 32. Thruster Calibration Data

3. Maxon Motor and EPOS 24/1 Positioning Controller

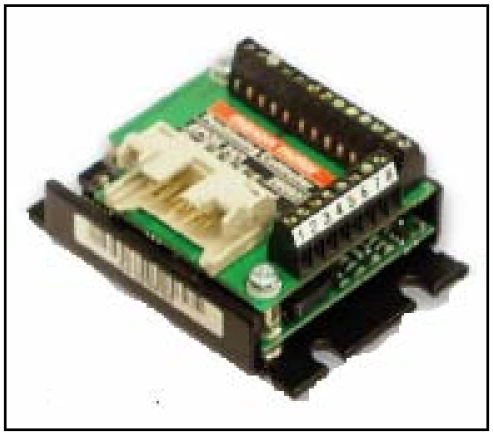


Figure 33. Maxon Motor EPOS 24/1 Positioning Controller (Ref. [18])

The Maxon Motor EPOS 24/1 positioning controller, shown in Figure 33, is used to control the brushless DC motors used to manipulate the MSGCMG and vectorable thrusters. The EPOS 24/1 for the MSGCMG is mounted on top of middle module deck while the two EPOS 24/1s for the dual vectorable thrusters are located underneath the deck of the middle module. The XPC Target control computer communicates with the EPOS 24/1 thru a RS-232 serial communication. Figure 34 illustrates the wiring schematic needed to setup the EPOS 24/1. Terminals 12 and 13 on the J1 connector are used to supply + 24 VDC and -24 VDC power respectively. This power comes from the 24 VDC power terminal board located underneath the middle module, as shown in Figure 10. A toggle switch, located on the middle module deck, is used to interrupt power to the EPOS 24/1s separately from the rest of the spacecraft simulator. It is necessary to remove power from the DC motors when repositioning them by hand to prevent damage to the motor or gears. Also, in order to establish the 'zero position' for each actuator, power must be removed, the actuator repositioned, and power then restored.

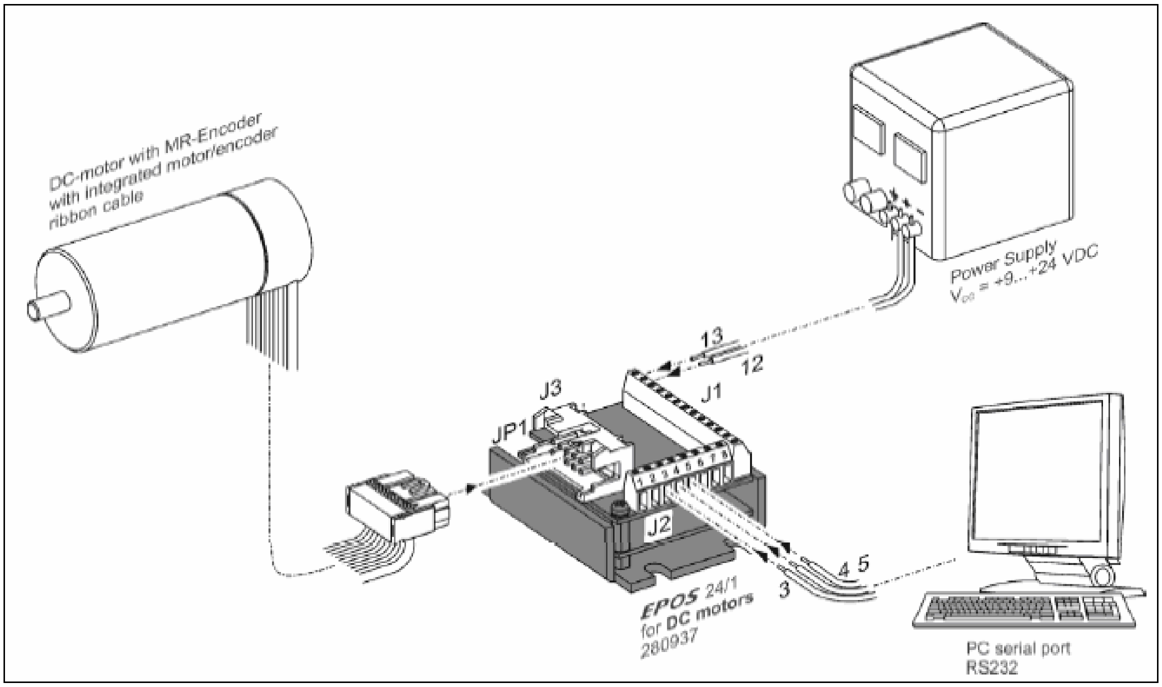


Figure 34. Maxon Motor EPOS 24/1 Positioning Controller Wiring Schematic (Ref. [18])

Each EPOS 24/1 uses three wires to establish its RS-232 serial communication. Terminals three, four, and five on the J2 connector correspond to the serial wires RXD, TXD, and GND respectively. The three serial wires from each EPOS 24/1 are combined into one wire bundle with a standard nine pin serial connector, allowing for easy wire disconnection between the spacecraft simulator modules. The opposite end of the combined wire bundle splits the individual EPOS 24/1 serial wires into separate nine pin serial connectors, labeled **A**, **B**, and **C** to allow for connection to the XPC Target control computer's COM ports one, two, and four respectively. The wiring scheme used to establish the serial connections to each of the EPOS 24/1s is listed in Table 11.

EPOS CONNECTION	**WIRE COLOR**	**WIRE BUNDLE PIN NUMBER**	**COM PORT PIN NUMBER**
MSGCMG EPOS (COM 1)			***Connector A***
3 (RXD)	Red	Pin 2	Pin 3 (TXD)
4 (TXD)	Orange	Pin 1	Pin 2 (RXD)
5 (GND)	Black	Pin 3	Pin 5 (GND)
Vectorable Thruster #1 (COM 2)			***Connector B***
3 (RXD)	Yellow	Pin 5	Pin 3 (TXD)
4 (TXD)	Blue	Pin 4	Pin 2 (RXD)
5 (GND)	White	Pin 9	Pin 5 (GND)
Vectorable Thruster #2 (COM 4)			***Connector C***
3 (RXD)	Green	Pin 6	Pin 3 (TXD)
4 (TXD)	Purple	Pin 7	Pin 2 (RXD)
5 (GND)	Gray	Pin 8	Pin 5 (GND)

Table 11. Nine Pin Connector Wiring Scheme

The data used to communicate with the EPOS 24/1s is transmitted asynchronously with one start bit, eight data bits, no parity and one stop bit. Each data byte is transmitted in sequential frames, shown in Figure 35, with each frame consisting of a header, a variably long data field and a 16-bit cyclic redundancy check (CRC) to verify data integrity, all in hexadecimal format. The header is a one word (16-bit) value composed of an 8-bit operation command (OpCode) followed by an 8-bit value that represents the number of words in the data field minus one. The data field contains the parameters of the particular message and is separated into groups of 16-bit words. The final component of the frame structure, the CRC, is a 16-bit word that must be calculated. In calculating the CRC, the code looks first at the header values followed by the high byte and then the low byte of each sequential data value followed finally by 0x0000. Before transmitting, the string must be reorganized to place the low bytes first followed by the high byte for each data word and the CRC word. (Ref. [18])

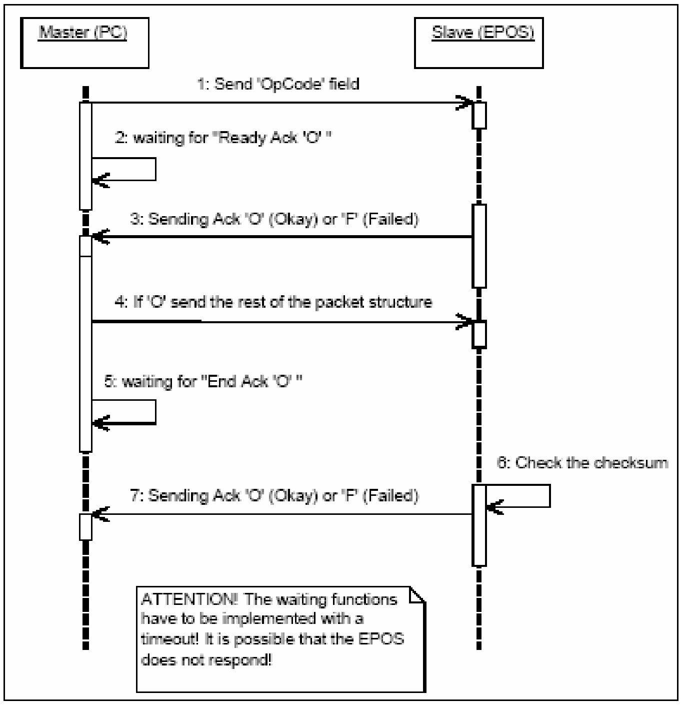

Master (PC)
Slave (EPOS)
1: Send 'OpCode' field
2: waiting for "Ready Ack 'O' "
3: Sending Ack 'O' (Okay) or 'F' (Failed)
4: If 'O' send the rest of the packet structure
5: waiting for "End Ack 'O' "
6: Check the checksum
7: Sending Ack 'O' (Okay) or 'F' (Failed)
ATTENTION! The waiting functions have to be implemented with a timeout! It is possible that the EPOS does not respond!

Figure 35. EPOS 24/1 Command Frame Structure Diagram (Ref. [18])

MATLAB to communicate with the EPOS 24/1 was developed prior with the development of the MSGCMG. This code development is described in Ref. [8]. This MATLAB code, however, was not able to be compiled for use in the real-time XPC Target application. Therefore, the MATLAB code was converted to a SIMULINK model with embedded MATLAB functions, as shown in Figure 36. The SIMULINK model utilizes the XPC Target toolbox RS-232 blocks to establish communications with the individual COM ports of the XPC Target computer. The model is divided into two enable function blocks, set to execute sequentially. The first function block to execute initializes the EPOS 24/1 during the first second of the simulation while the second function block executes during the rest of the time and is used to send the velocity and report position commands to the EPOS 24/1 and receives the responses. The received EPOS 24/1 response is sent thru an embedded MATLAB function, listed in Appendix F, which parses out and interprets motor position information in degrees. This is then converted to radians for use as feedback to the particular actuator control logic using that EPOS 24/1. This application will be further described in proceeding sections.

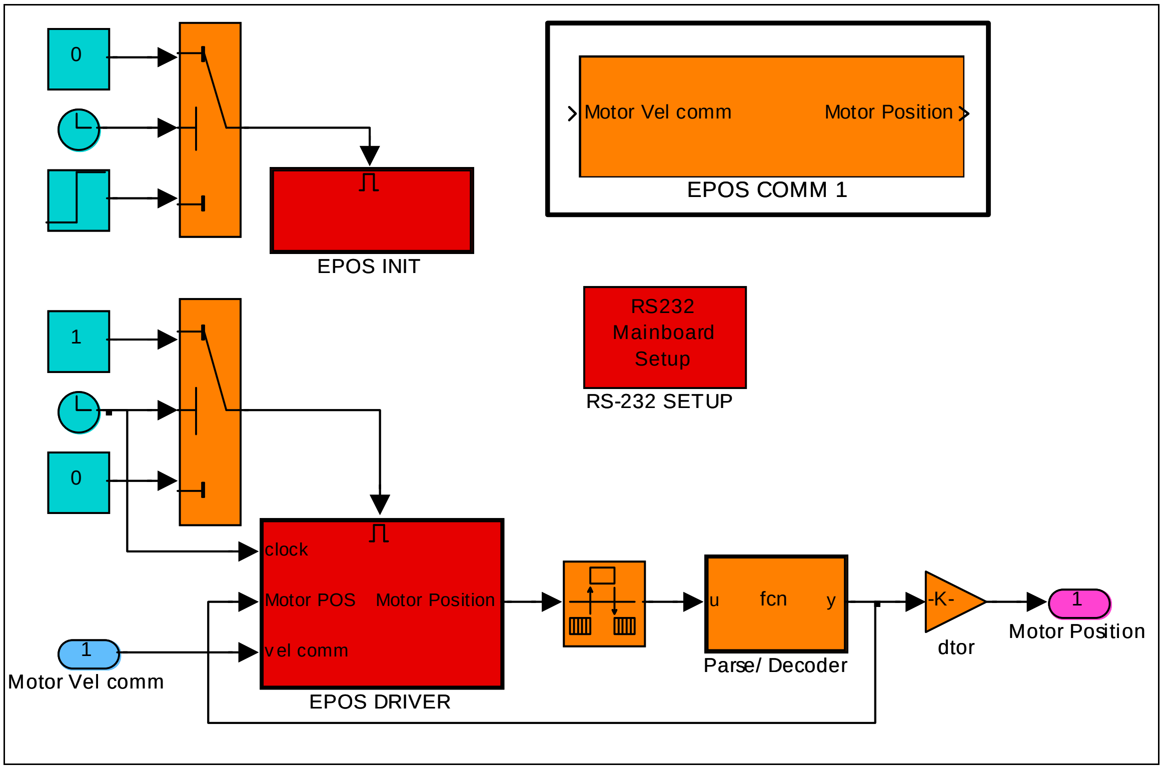


Figure 36. EPOS 24/1 Position Controller SIMULINK Model

The EPOS 24/1 initialization function block, shown in Figure 37, consists of a series of if action enable function blocks. These must be used to switch between different RS-232 send blocks. The RS-232 send block in the XPC Target toolbox only allows a fixed length of bytes to be sent to a COM port, a separate block must be used for each different command length. Command lengths can either be a single byte, seven bytes, or eleven bytes in length. Since the commands that need to be sent to initialize the EPOS 24/1 are known and constant, constant blocks can be used to send them. Repeating sequence blocks are used to cycle thru the different commands and the required RS-232 block that is setup for the length of the particular command. The initialization command sequence clears all faults, enables the EPOS 24/1, sets appropriate profile parameters on the controller, and sets the EPOS 24/1 to be in velocity mode. The decimal command sequence and its purpose are listed in Appendix F.

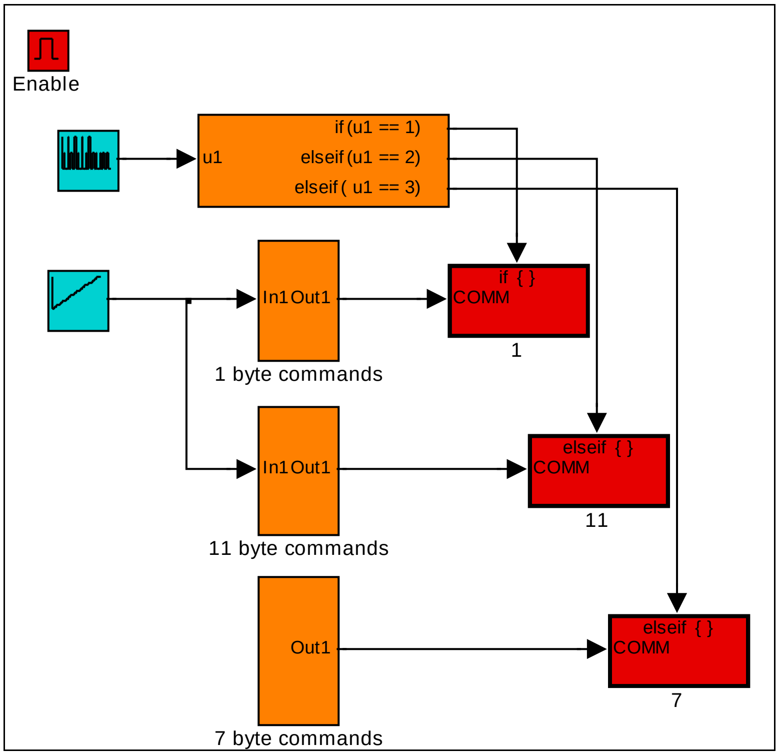


Figure 37. EPOS 24/1 Initialization SIMULINK Sub Model

The function block used for normal communications with the EPOS 24/1, shown in Figure 38, is set up very similar to the initialization function block. A velocity command and a command for the EPOS 24/1 to report its current position are sent. The command sequence, listed in Appendix F, is sent in a continuously repeated loop at a frequency of 25 Hz with only the velocity command changing over time. The required velocity signal, sent from the controller, enters the function block in radians per second where it is then converted to degrees per second. The signal is then sent to an embedded MATLAB function, listed in Appendix F, which encodes the proper EPOS command to be sent. A time controlled switch is used to re-center the DC motor just prior to the end of the simulation run time.

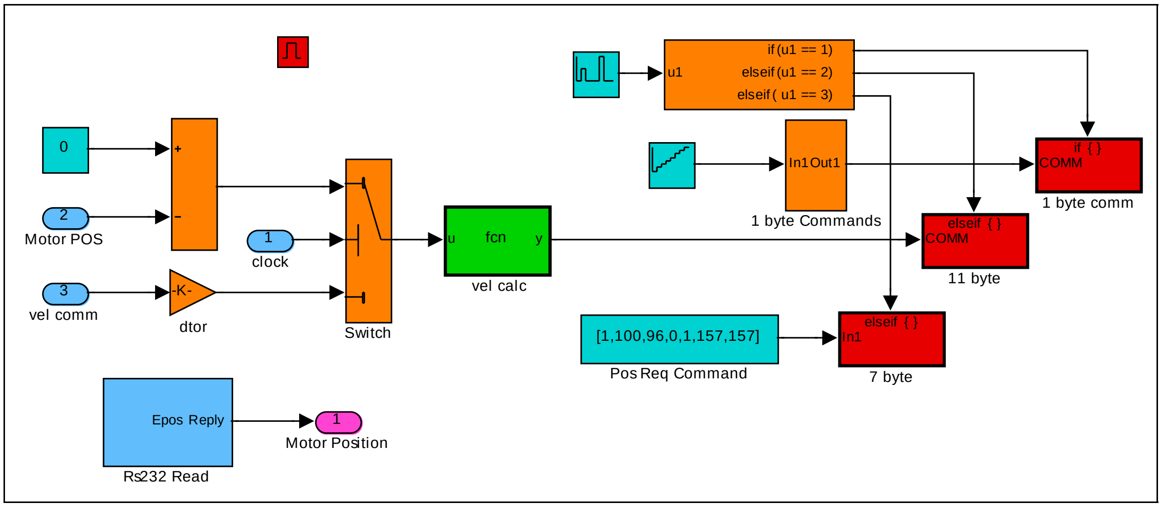


Figure 38. EPOS 24/1 Driver SIMULINK Sub Model

THIS PAGE INTENTIONALLY LEFT BLANK

III. CONTROLLER AND ACTUATOR MAPPING ALGORITHM DESIGN

A. SPACECRAFT SIMULATOR DYNAMICS

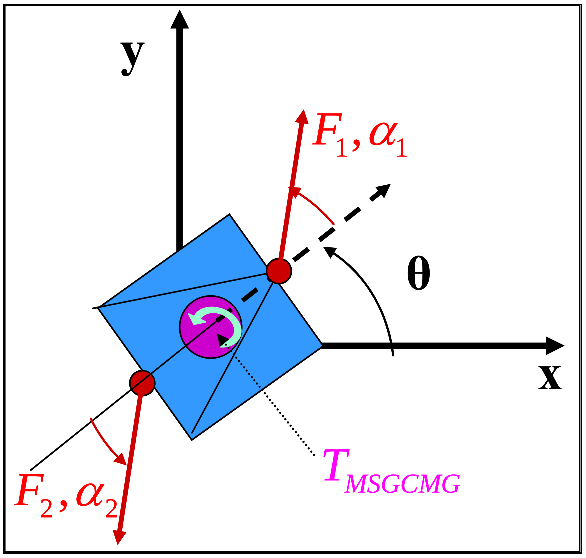


Figure 39. Spacecraft Simulator Dynamics Schematic

The first step in developing a controller is to determine the dynamics for the spacecraft simulator. Figure 39 illustrates the reference frames used in developing the parameters to describe the spacecraft simulator's orientation and the effect of the individual actuators. The spacecraft simulator has two degrees of translational freedom, described with coordinates X and Y, and one degree of rotational freedom about the third Z-axis, described with the coordinate θ.

There are five control parameters that affect the spacecraft simulator. The first one is the torque generated by the MSGCMG, referred to by T_{MSGCMG}. The remaining control parameters describe the effect of the dual vectorable thrusters. Each thruster as a force magnitude parameter, F_1 or F_2, and direction, α_1 or α_2, determined by the slew angle of the individual vectorable thruster. The slew angle for each vectorable thruster is measured from an axis perpendicular to the faces of the spacecraft simulator that the particular thruster is mounted on with a right hand rule convention, as shown in Figure 39. Thruster one is mounted on the front of the spacecraft simulator, while thruster two is mounted on the opposite face. Equations(3),(4), and (5) describe the dynamics associated with the spacecraft simulator. The fact that there are more control parameters than there are dynamic equations and the non-linearity of the dynamic equations adds complexity to developing an actuator mapping algorithm.

$$\ddot{x} = F_1 \cos(\alpha_1 + \theta) - F_2 \cos(\alpha_2 + \theta) \tag{3}$$

$$\ddot{y} = F_1 \sin(\alpha_1 + \theta) - F_2 \sin(\alpha_2 + \theta) \tag{4}$$

$$\ddot{\theta} = T_{MSGCMG} + F_1 d_1 \sin(\alpha_1) + F_2 d_2 \sin(\alpha_2) \tag{5}$$

B. CONTROLLER DESIGN

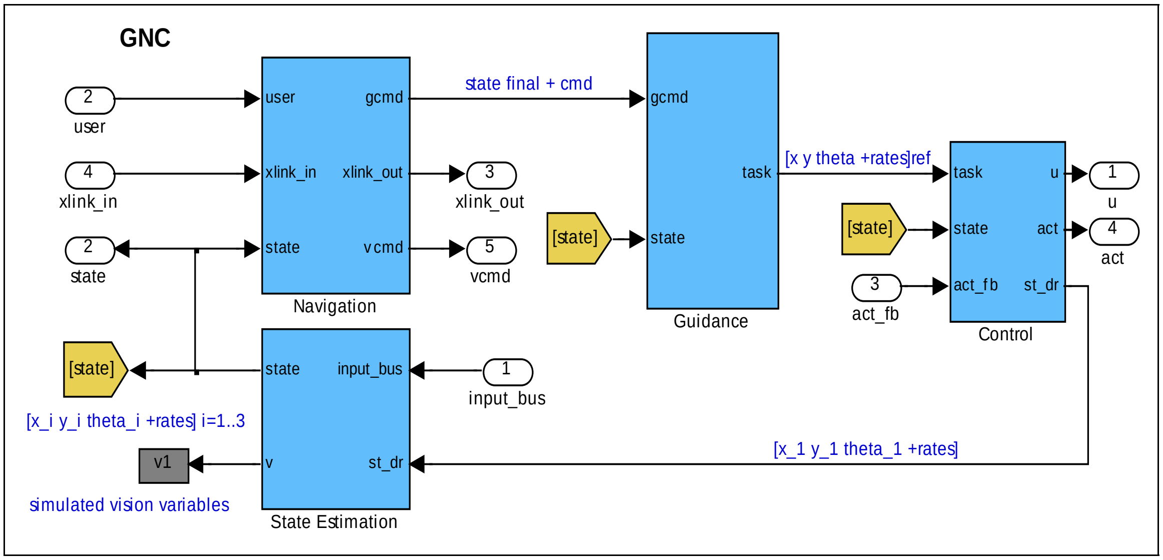


Figure 40. Guidance, Navigation, and Control SIMULINK Model (Ref. [9])

To compliment the modularity of the hardware design of the spacecraft simulator, the software architecture that was developed, particularly for the XPC Target control computer, also is modular in design. This is easily accomplished with the graphical block design of SIMULINK. The controller function block is a part of the Guidance, Navigation, and Control block, shown in Figure 40, of the overall SIMULINK model shown in Figure 15. Further discussion of the modular design of the overall software architecture can be found in Ref. [9].

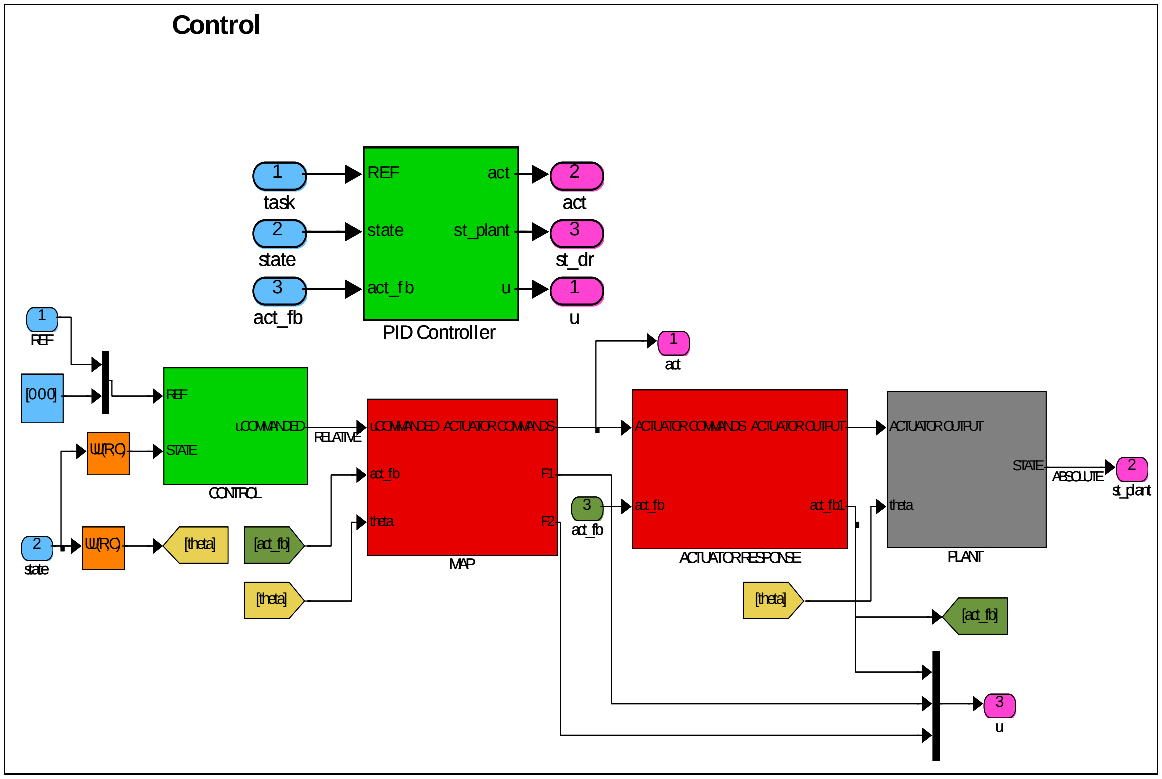


Figure 41. Control and Mapping SIMULINK Model

The control and mapping function block, shown in Figure 41, is divided in four parts. The first two, the controller and mapping blocks, are used to generate a required control signal and then map that signal to command signals that can be sent to the actuator blocks described in the previous section. The SIMULINK model developed to control the spacecraft simulator can also be used as a computer simulator to test various guidance and control algorithms. Therefore two addition blocks, actuator response and plant, were developed. These two blocks simulate the response of the actuators for the given actuator commands and the effect of those actuator responses on the overall spacecraft.

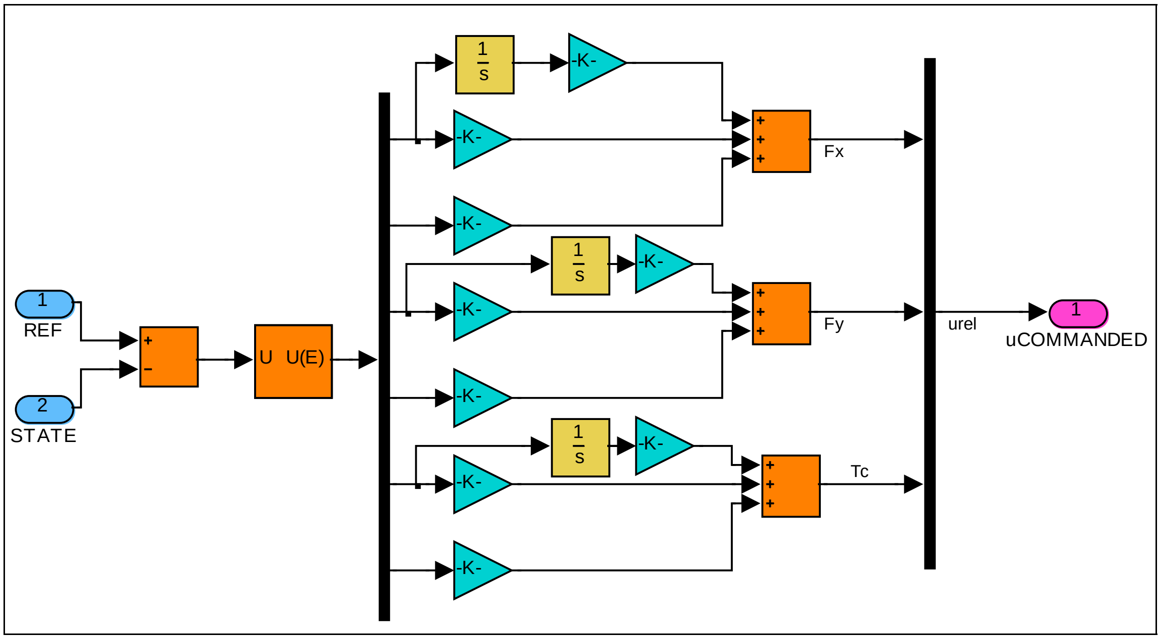


Figure 42. Controller SIMULINK Model

As an initial step in the development of the AMPHIS test bed, a generic proportional-integral-derivative (PID) controller was developed, shown in Figure 42. This controller compares the reference signal state, which consists of $\left[x, y, \theta, \dot{x}, \dot{y}, \dot{\theta}\right]$, with the current state as determined by the state estimator. The functionality of the guidance system and state estimator is discussed in Ref. [9]. The controller generates a required absolute force signal of F_x, F_y and T. These absolute required force signals are used by the mapping block to generate required relative actuator commands to affect the required absolute forces. The gains for the PID controller were adjusted so that the system is slightly over-damped, minimizing the possibility of overshoots resulting in a collision.

C. ACTUATOR MAPPING ALGORITHM DEVELOPMENT

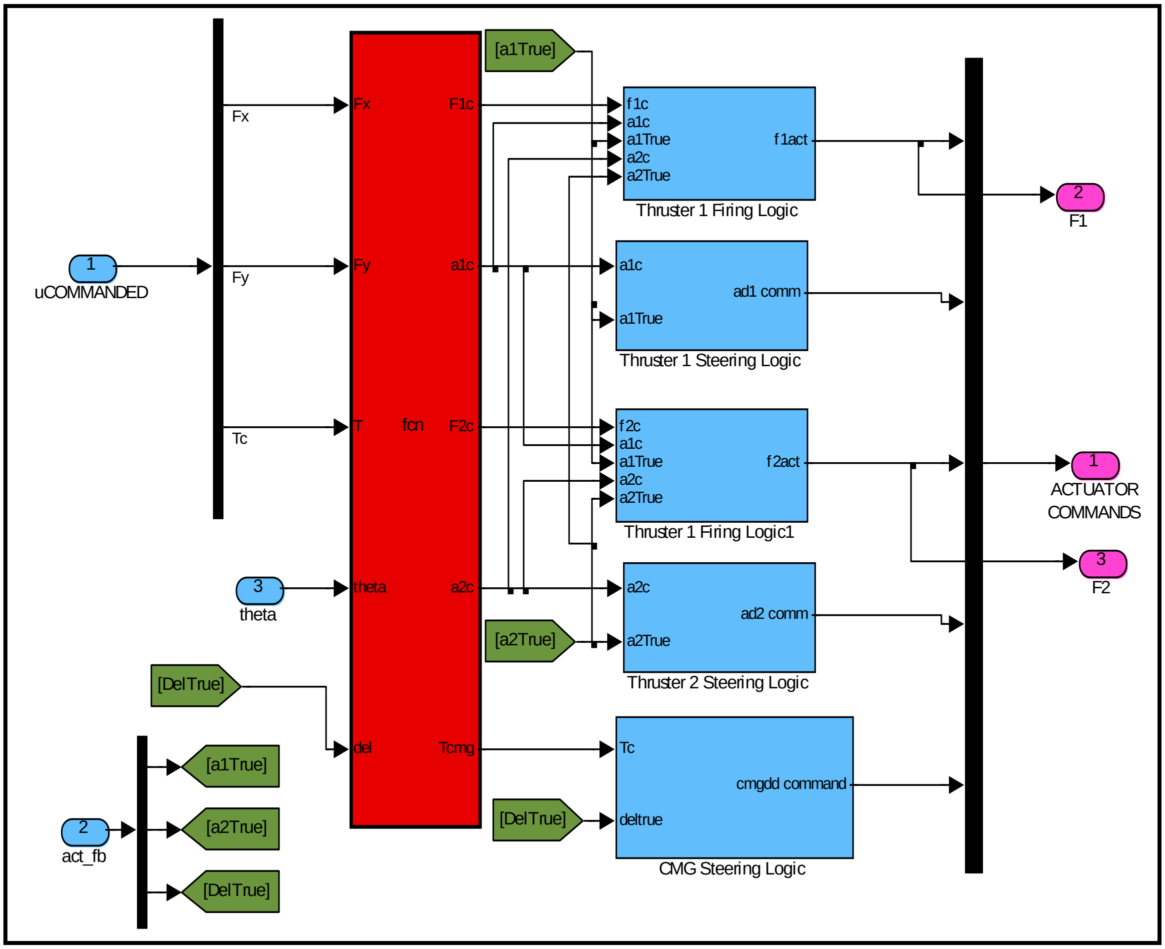


Figure 43. Actuator Mapping Algorithm SIMULINK Model

The actuator mapping algorithm block, shown in Figure 43, takes the required absolute forces generate by the controller, along with current attitude and MSGCMG position information, and maps these to relative commanded output for each actuator. Individual actuator command logic blocks take the required actuator outputs and develop a command signal that can be sent directly to the actuator control blocks.

Due to the multiple available control parameters, a method must be used to designate which actuator performs which tasks. To resolve this issue, a master/slave approach was adopted. The torque parameter generated by the MSGCMG operates as the master controller for attitude. The required absolute force signal is translated directly to a relative force vector for each vectorable thruster. Leveraging the ability of the thrusters to slew and fire independently, MSGCMG position is used to generate a difference signal between the two thrusters. This difference signal applies a torque about the spacecraft to drive the MSGCMG back to the neutral position. This has the benefit of not only desaturating the MSGCMG if it should get saturated, by also of applying a torque to assist the MSGCMG as soon as the MSGCMG moves from the neutral position to respond to a required torque signal. The embedded MATLAB function to perform these functions is found in Appendix G.

The individual actuator logic blocks convert the required actuator operation to an actuator command signal. For the thruster firing logic, this is done by ensuring that both vectorable thrusters are aligned in their commanded positions prior to generating a fire thruster command. The thruster slewing logic converts the required thruster position command and converts it to a thruster slew velocity based on current thruster position received from the actuator feedback. The MSGCMG logic block converts the torque command in a MSGCMG slew velocity.

D. ACTUATOR AND PLANT RESPONSE MODEL

To allow for the ability to use the SIMULINK software developed for the XPC Target control computer as a computer simulation model as well, actuator and plant response models had to be developed. The goal is to have the signals generated by these models to emulate the signals and responses that would be generated when the actual hardware is being used. The actuator response model, shown in Figure 44, receives the actuator command signals generated by the mapping model, and generates a response that is similar to the response that the actual actuators would generate, including the motor position feedback response of the EPOS 24/1s. For the MSGCMG, the velocity command is converted by to a torque signal. For the vectorable thrusters, the digital thruster fire signal is converted to a relative force. A manual switch is used to select whether the simulated actuator response or the actual actuator response is used by the rest of the software.

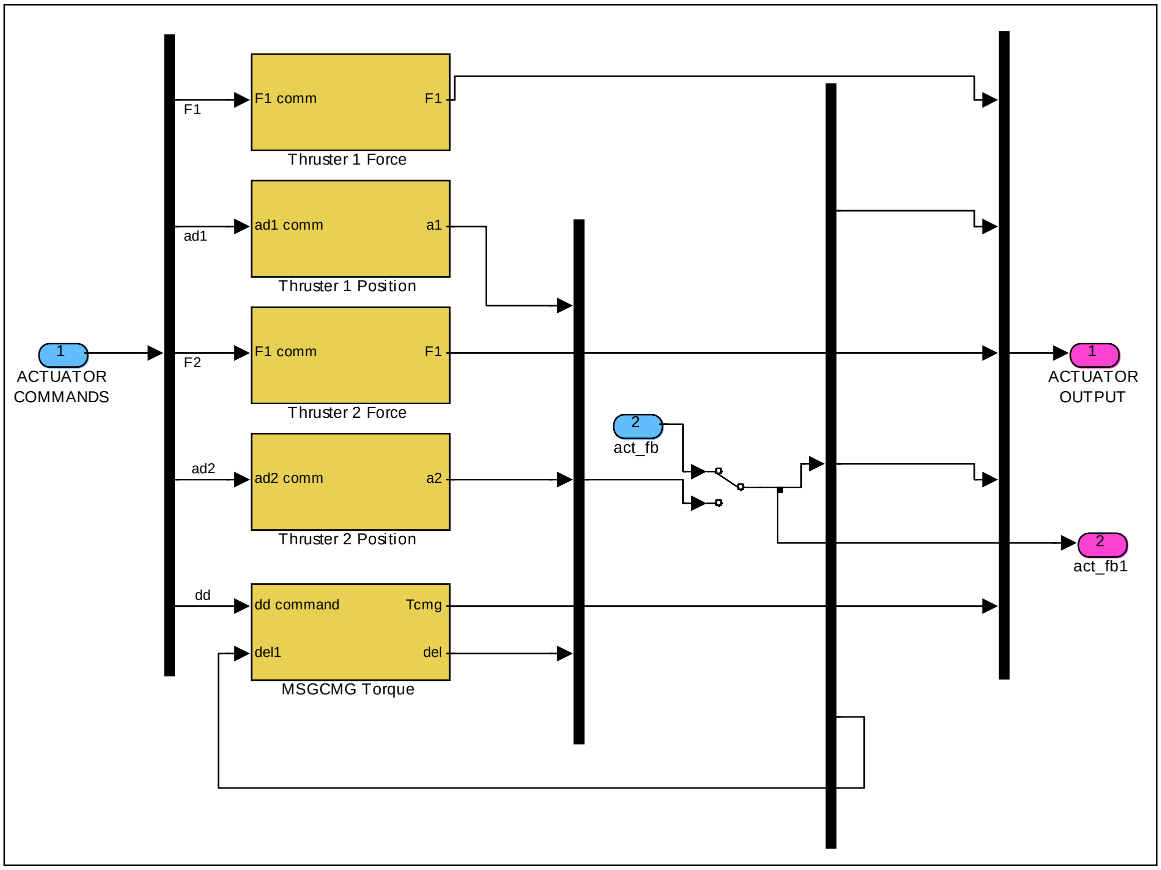


Figure 44. Actuator Simulated Response SIMULINK Model

The plant response model, shown in Figure 45, uses actuator response signals (either simulated or actual) and simulates the dynamics and kinematics of the physical spacecraft simulator. The first block converts the actuator response signals and current attitude to convert the relative force to absolute forces. These absolute forces are used by a plant simulator, using the mass of moment of inertia characteristics of the actual spacecraft simulator, to propagate the spacecraft simulators response in the inertial frame. The state estimator block contains a manual switch that allows for the selection of either the plant response model generated state or the actual information from the onboard sensors to use as the state information by the rest of the SIMULINK model. The plant state information may also be used in the future by the state estimator as part of a Kalman Filter.

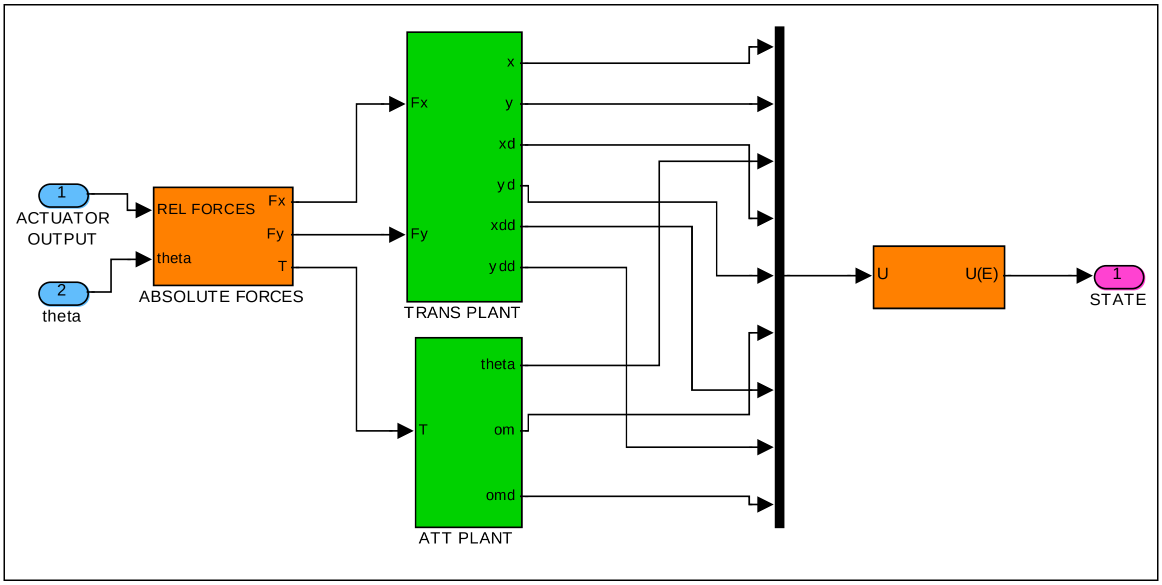


Figure 45. Physical Plant SIMULINK Model

IV. CONCLUSION

A. RESULTS

A sample maneuver was used to verify the effectiveness of the controller, mapping algorithm, as well as the rest of the hardware and software architecture that was developed for the AMPHIS test bed. The sample maneuver of moving to a point three meters in the x direction and 2 meters in the y direction, with a 90 degree rotation, was used to test the spacecraft computer simulator as well as hardware in the loop with the AMPHIS experimental test bed.

1. Computer Simulation Results

The computer simulation results, shown in Figure 46, demonstrate the effectives of the controller and actuator mapping algorithm. The overall system response was slightly over-damped, with the spacecraft simulator arriving at its target position in approximately 95 seconds. The maneuver required 25.34 Ns of thrust from the vectorable thrusters.

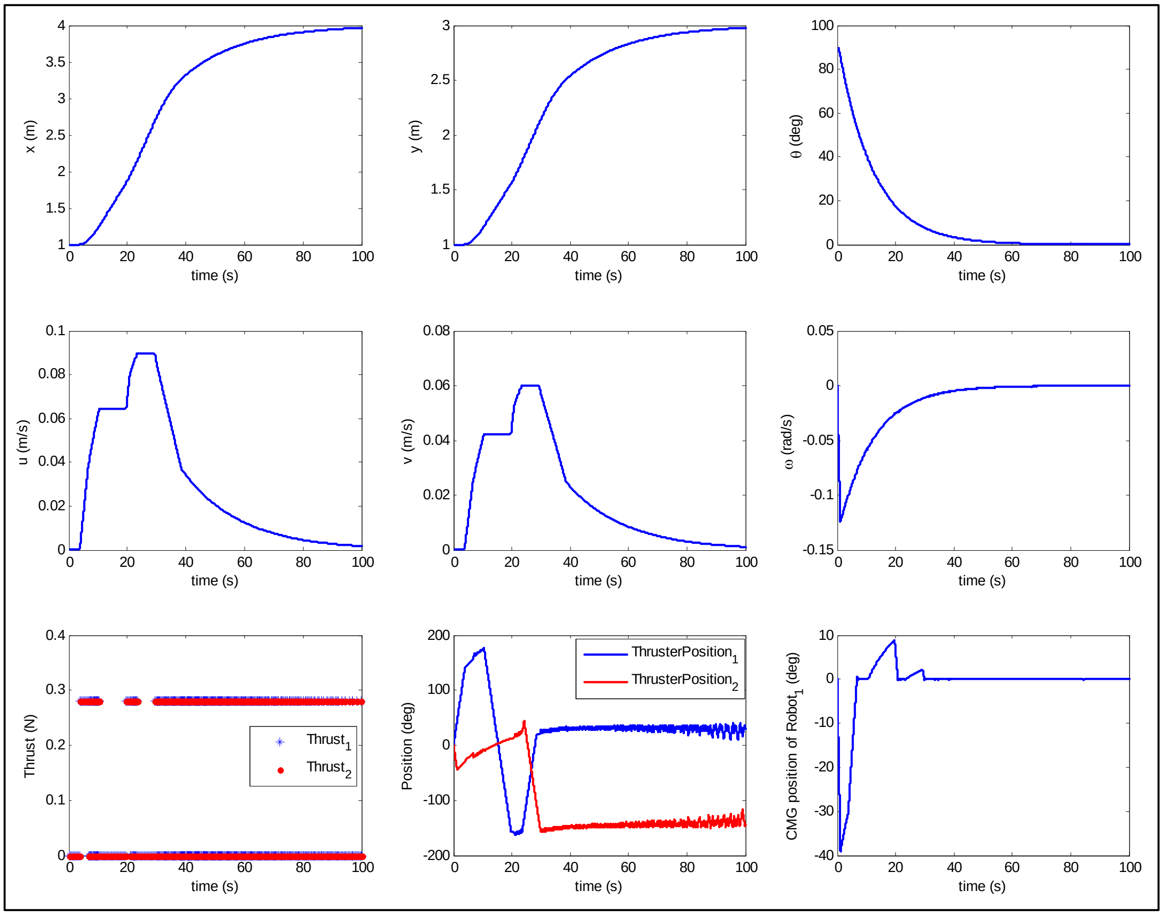


Figure 46. Simulation Results

2. Hardware in the Loop Experimentation Results

The hardware in the loop experimentation results, shown in Figure 47, demonstrate the effectives of the controller and actuator mapping algorithm to work on an actual three DOF test bed. The overall system response was very similar to the result obtained from the computer simulation, with the spacecraft simulator arriving at its target position in approximately 98 seconds. The maneuver required 32.56 Ns of thrust from the vectorable thrusters. There was more noise in the actual thruster and MSGCMG positions. This added noise, however, did not significantly affect the spacecraft simulators performance. The relative similarity between the simulation data and the experimentation data adds the confidence that the simulation models being used are realistic.

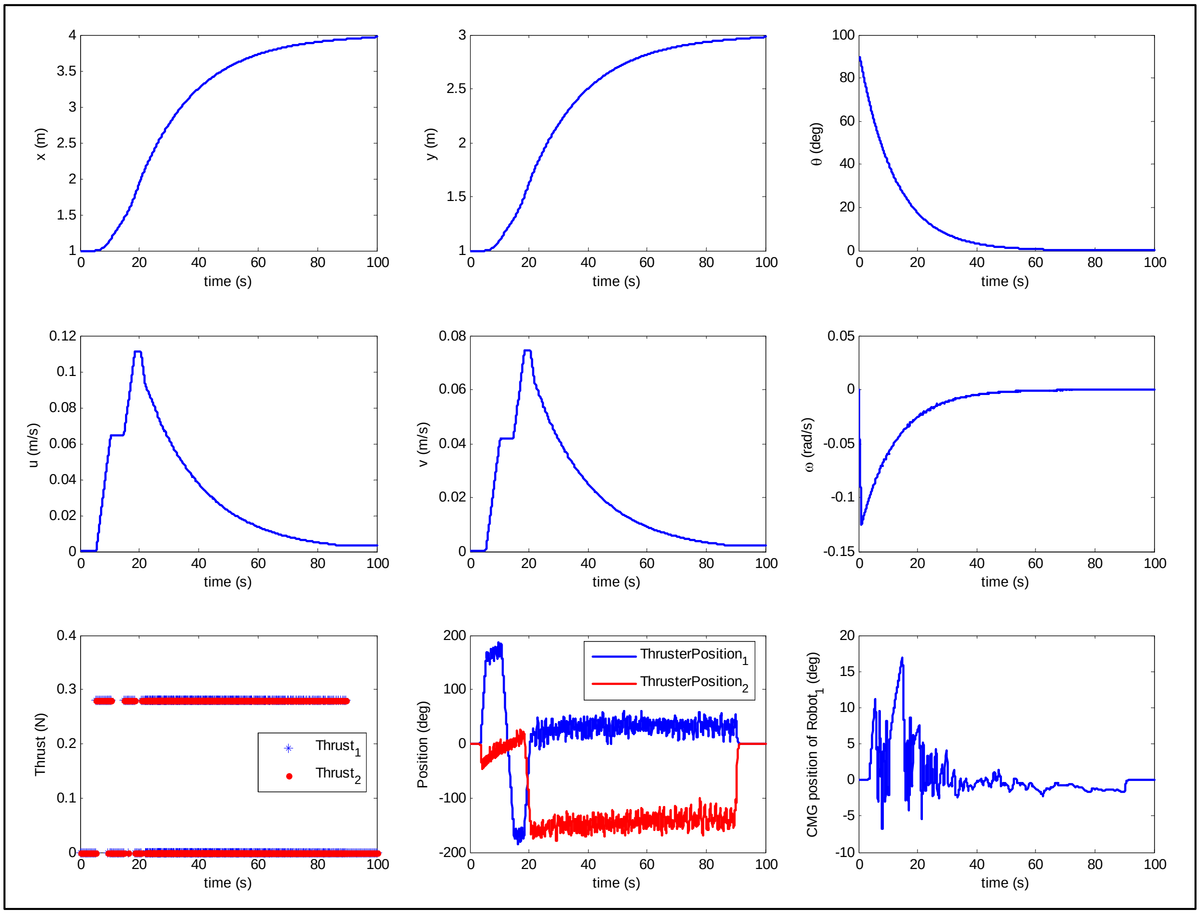


Figure 47. Hardware in the Loop Experimentation Results

B. FUTURE WORK

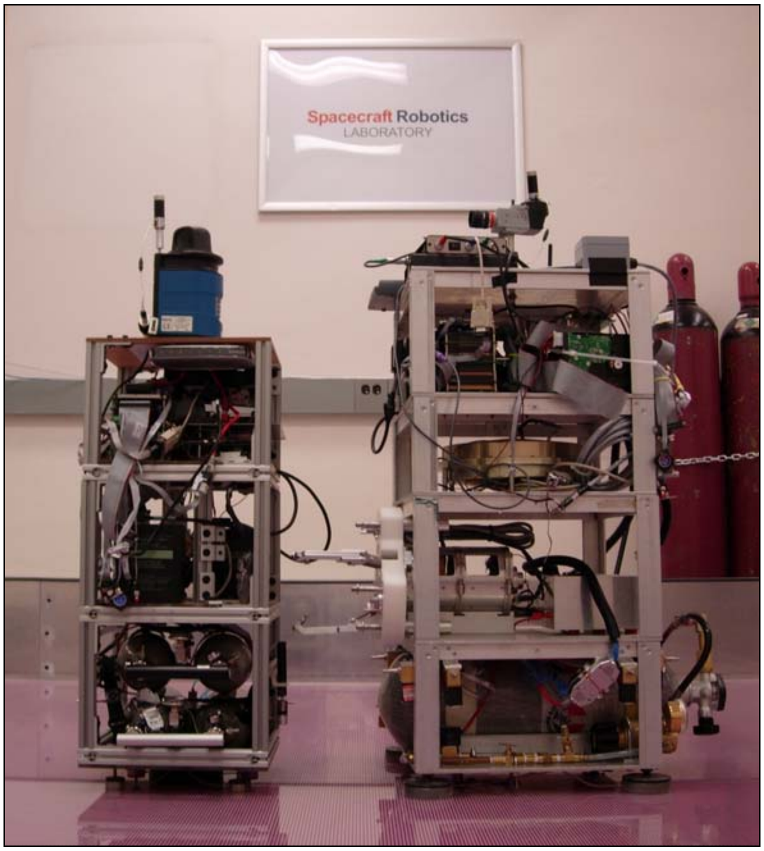


Figure 48. AMPHIS Rendezvousing with AUDASS

Further development of the AMPHIS test bed still needs to be completed. This includes the addition of docking mechanisms and the completed integration of the LiDAR sensor information into the state estimator. Also, the construction of additional spacecraft simulators needs to be completed to allow for true multi-agent spacecraft testing. One of the systemic problems encountered with the spacecraft simulator hardware is the wear on the air pads used in the floatation system. The addition of a sub micron air filter to the floatation should help alleviate this issue.

Besides further hardware improvements, more robust and efficient controllers can now be developed and tested on the AMPHIS test bed. Also, different sensor and actuators can be developed and/or tested on the AMPHIS spacecraft simulator bus. Truly the AMPHIS test bed and POSF are in a condition were new and exciting autonomous multi-agent spacecraft controller design, development and testing can be done.

APPENDIX A. COMPONENT MANUFACTURER AND LIMITATION INFORMATION

A. STRUCTURAL COMPONENTS

Aluminum T-Slotted Framing System:

Manufacturer: 80/20
Vendor: McMaster-Carr
http://www.mcmaster.com/

Static Dissipative Rigid Plastic Sheets (12 x 12 x ¼ inch)

Vendor: McMaster-Carr

B. FLOATATION SYSTEM

Air Cylinders:

Manufacturer: Pure Energy paintball bottle, Model #40669
Vendor: Palmers Pursuit Shop
(http://www.palmer-pursuit.com)
Limitations: 3000 PSI, 68 cu in per
QD Style fill fitting

Stabilizer Dual Manifold System:

Manufacturer: Palmers Pursuit Shop
Limitations: Dual paintball bottles
Regulated pressure between 0 and 300 PSI

1/8 in OD- 1/16 in ID Low-Pressure Poly Tubing:

Vendor: Palmers Pursuit Shop
Limitations: 500 PSI

1/8 NPT Brass Fittings (Various):

Vendor: Palmers Pursuit Shop
Limitations: 500 PSI

Solenoid

Manufacturer:	ASCO, Model #U 8225B002V
Limitations:	24 VDC, Normally Closed, rated for fluid and gas Maximum pressure of 125 PSI

32 mm Air Bearings:

Manufacturer:	Aerodyne Belgium, PERARA Dextair (PE032)
Vendor:	Ameropean (No longer distributing)
Limitations:	Maximum Loading of 125 N @ 4 bar ensures 10 micron air gap

C. POWER DISTRIBUTION SYSTEM

Lithium Ion Battery Packs:

Manufacturer:	UltraLife Batteries, Inc., Model #UBBL02 http://www.ultralifebatteries.com/
Limitations:	28 V for 6 AH or 14 V for 12 AH per battery pack

DC-DC Converter Array:

Manufacturer:	Vicor, Standard VIPAC Array http://www.vicr.com/
Limitations:	Input Voltage 24 VDC Max power output per converter 100 W

Mechanical Relay Array:

Manufacturer:	RTD Embedded Technologies, Model # DMR8 http://www.rtdusa.com
Limitations:	±5 VDC operating Capable of supporting 8 separate devices

D. ACTUATORS

Thruster Solenoids:

Manufacturer:	Precision Dynamics, Inc., Model # EH2012-C204

http://www.predyne.com

Limitations: +24 VDC, Normally Closed, rated for fluid or gas
3-5 milliseconds switching capability

Thruster Nozzles:

Manufacturer: Silvent, Model # MJ5
http://www.silvent.com

Limitations: 5.9 scfm air consumption, 1.8 N force with 72 PSI supply, M5x.5 connection

MSCMG gyroscope:

Manufacturer: Educational Innovations, Inc., Super Motorized Precision Gyroscope
http://www.teachersource.com

Limitations: ±5 VDC, 12000 rpm rotor wheel rotation rate

MSCMG gimbal motor and Vectorable Thruster slew motor:

Manufacturer: Maxon Motor USA,
RE16 motor, Model # 118730
MR encoder, Model # 201940
EPOS 24/1 Positioning Controller

Limitations: +9 to +24 VDC, RS-232 serial interface 1 kHz Update capability
Max rotation rate of 16000 rpm
Max continuous current of .614 A
Max torque $4.98x10^{-3}$ Nm
Torque Constant of $8.11x10^{-3}$ Nm/A

E. SENSORS

LiDAR Sensor:

Manufacturer: SICK AG, Model # LD-OEM 1000
http://www.sick.com/home/en.html

Limitations: +24 VDC ± 20%, RS-232 serial interface

iGPS:

Manufacturer: Metris
http://www.metris.com/

Limitations: Range between 2 and 40 m
Requires Windows XP based PC running
Workspace software

Three-Axis Accelerometer:

Manufacturer: Crossbow, Model # CXL02TG3
http://www.xbow.com/

Limitations: +5 VDC, 2 pin in and 4 pin out (3 axis and temp)
$8.5x10^{-3}$ g bias stability and ± 2 g input range

Single Axis Fiber-Optic Rate Gyro:

Manufacturer: KVH Industries, Inc., Model # DSP-3000
http://www.kvh.com/

Limitations: Digital, 100 Hz asynchronous communication via
RS-232 interface at 38,400 baud, + 5 VDC, input
rate up to ± 375 deg/sec, Offset bias ± 20 deg/hr

F. COMMAND AND DATA HANDLING

108 Mbps Wireless Router:

Manufacturer: Netgear, Model # WGT624 v2
http://www.netgear.com/

Limitations: + 12 VDC
Provides both wired and wired TCP/IP routing

Pentium III PC-104 Computer:

Manufacturer: Versalogic Corporation
http://www.versalogic.com/

Limitations: + 5 VDC

APPENDIX B. PROCEDURES FOR SETTING UP THE XPC TARGET CONTROL COMPUTER

The following details the setup procedures for the Jaguar Pentium III CPU to include installation of the necessary MATLAB XPC Target code for the spacecraft simulators associated with the AMPHIS test bed.

A. REQUIRED COMPONENTS LISTS

1. Versalogic Jaguar Pentium III PC-104 computer
2. 256 MB SDRAM Module
3. 96 MB Disk on Chip (DOC)
4. IDE, KVM, and VGA ribbon connector cables
5. PC-104 power supply
6. Diamond Systems DMM-32X-AT Analog I/O PC-104 Module
7. Versalogic Quad RS-232 Module
8. 3.5” Floppy Disk Drive
9. DOS 6.22 3.5” Setup Disk
10. Blank 3.5” disk
11. Desktop with Matlab2006b with Realtime Workshop Toolbox and Visual C++ installed.
12. ATX power supply

B. SETUP PROCEDURE

1. Start MATLAB2006b residing on the desktop PC.
2. At the prompt in the MATLAB command window, type xpcexplr and then expand the TargetPC icon in the XPC Target Hierarchy on the left of the screen.

3. Under the Communication icon select TCP/IP as the Host target communication under Communication protocol. Under the Target PC TCP/IP configuration, use the following:

   a. Target PC IP Address: 192.168.1.x13, where x designates the spacecraft simulator number. (Refer to Table 3)

   b. TCP/IP target driver: I82559

   c. TCP/IP target port: 22222

   d. TCP/IP target bus: PCI

   e. LAN subnet mask address: 255.255.255.0

   f. TCP/IP gateway address: 255.255.255.255

4. Under the Settings icon, set Target RAM size (MB) to Auto and select 16 MB as the maximum model size. Leave all other boxes unchecked.

5. Under the Appearance icon, check the Enable target scope and select none for the target mouse.

6. Insert a blank 3 1/2 inch floppy disk into the desktop PC and then under the Configuration icon, select DOSLoader and then click Create Bootdisk.

7. Remove the XPC Target DOSLoader disk from the desktop computer.

8. Install DOC and SDRAM onto the PC-104 mother board.

9. Setup the Quad RS-232 Module by setting the interrupts for COM ports three are four to IRQ five and 10 respectively.

10. Setup the DMM-32X-AT Analog Module by setting jumper J5 pins one and two in and pins three, four, five, and six to out. Also, set jumper J6 connection five to in and connections 10, P, and B to out. Connection R can be set either way.

11. Construct the PC-104 stack with the Jaguar computer, a PC-104 power supply, DMM-32X-AT module, and the Quad RS-232 module.

12. Attach connector cables to PC-104.

13. Attach 3.5” floppy disk drive. Insert DOS 6.22 floppy disk.

14. Attach ATX power supply to PC-104 and floppy disk drive.

15. Power up the computer. Press ‘delete’ during startup to enter the system CMOS.

16. Verify that the A:\ drive is listed as one of the bootable devices and that the C:\ is not enabled in the Basic Settings. Under the Custom settings, disable the integrated COM ports by setting COM one and two to disabled. Also, enable the DOC by assigning it a memory address. The DOC will act as the C:\ if there is not HDD installed.

17. Reboot. The computer should boot of the 3.5” DOS disk. Verify that the C: drive is available. At the A:\ prompt, format the C: drive by typing ‘format c: /s’. This would load the system files onto the C:\ drive.

18. Copy DOS onto the C: drive by typing ‘xcopy a: c:’ at the a:\ prompt. Remove the 3.5” DOS disk.

19. Reboot. The computer should now boot off the C: drive (DOC) and load DOS. Insert the XPC Target DOSLoader disk. Copy the contents of the A: drive onto the C: drive by typing ‘xcopy a: c:’ at the C:\ prompt. Replace all duplicate files when prompted.

20. Remove the 3.5” disk. Reboot. The computer should now reboot and load the XPC Target Application. The XPC Target computer is now ready to be integrated into the AMPHIS test bed.

THIS PAGE INTENTIONALLY LEFT BLANK

APPENDIX C. PROCEDURES FOR SETTING UP WINDOWS XP BASED COMPUTER

The following details the setup procedures for the Jaguar Pentium III CPU with Windows XP to include installation of the necessary Windows applications.

A. REQUIRED COMPONENTS LISTS

1. Versalogic Jaguar Pentium III PC-104 computer
2. 256 MB SDRAM Module
3. 3.5” laptop HDD
4. IDE adapter the converts the female 40-pin HDD connection to a 44-pin 3.5” HDD connection with a five VDC power adapter.
5. IDE, KVM, USB, and VGA ribbon connector cables
6. PC-104 3.5” HDD mounting bracket
7. PC-104 power supply
8. Versalogic Quad RS-232 Module
9. CD-ROM drive
10. Windows XP disk
11. MATLAB 2006b installation disk
12. iGPS Workspace installation disk
13. ATX power supply

B. SETUP PROCEDURE

1. Install the SDRAM onto the Jaguar.
2. Build the PC-104 stack with the Jaguar, PC-104 power supply, the Quad RS-232 module, and the 3.5” HDD mounted on the PC-104 HDD bracket.
3. Attach IDE with adapter, KVM, VGA, USB and ATX power supply cables. Connect the CD-ROM.

4. Power on the computer. Hit the ‘delete’ key to enter the CMOS settings. Set the ATA DRV Assignment setting to ‘AUTOCONFIG, LBA’. Verify that Drive C: is set to IDE 0 and drive D: is set to IDE 1. Verify that CDROM is listed on the boot order.

5. Insert the Windows XP installation CD. Reboot the computer.

6. Install Windows XP.

7. Install MATLAB 2006b.

8. Install iGPS software per the instructions listed in Ref. [8].

9. Install a remote desktop software to remotely access the Windows XP computer.

10. Setup Windows XP account to boot up with user login.

11. Using Notepad, create a *.bat file to automatically load iGPS software in high priority mode. Use the following commands:

    ```
    cd c:\Program Files\Arc Second\WorkSpace\

    start /high workspace
    ```

    (ENSURE TO SAVE FILE WITH **.bat** extension.

12. The Windows XP computer is now ready to be integrated into the AMPHIS test bed.

APPENDIX D. PRE-EXPERIMENTATION SET-UP PROCEDURES

A. VEHICLE START-UP PROCEDURES

Prior to using the AMPHIS spacecraft simulator, ensure that both power cables are properly connected to the battery packs and that the vehicle is in a safe location with respect to the edges of the simulation floor. Additionally, ensure the air cylinders are filled to 3000 psi. The procedures for battery charging and refilling the air cylinders can be found in Ref. [8]. **BE CAREFULL OF STATIC CHARGE WHILE IN CONTACT WITH SPACECRAFT SIMULATOR ON THE EPOXY FLOOR. WILL CAUSE ALL COMPUTERS TO REBOOT!!**

1. Verify that all actuators (MSGCMG and both vectorable thrusters) are in their neutral or zero position.

2. Turn the simulator power switch to the ON position. All components should turn on, with the exception of iGPS.

3. After waiting approximately 2 minutes to allow the on-board Versalogic PC-104 to boot, energize the iGPS system by turning its power switch to on.

4. Connect to the wireless network for the AMPHIS prototype spacecraft simulator (AMPHISNET) from the demonstration computer.

5. Click on the Anyplace Control Admin Module desktop icon on the demonstration computer.

6. When the connection window appears, highlight the computer 192.168.1.2 by clicking on it and the click the connect icon. This will bring up a new screen from which you can control the on-board computer.

7. Establish communications with the iGPS receiver by turning on the receiver COM port in the iGPS WorkSpace application.

8. Build and load the XPC Target SIMULINK model (Robot.mdl) onto the XOC Target control computer. Verify that the manual switches are set for their intended positions. (Either simulation or hardware in the loop)

9. Run the Lynx based iGPS communications device on the Lynx computer.

10. Open the Windows XP computer SIMULINK model (windows.mdl) on the Windows XP computer.

11. If a hardware in the loop experiment is being executed, turn on the manual air valves to both the floatation and propulsion systems.

12. Run the Windows XP SIMULINK model. This model will execute the XPC Target control computer real-time application and retrieve all the data after the simulation and save it in a ***.mat** located in the XPCDatalog folder, named with the simulation run time.

APPENDIX E. DEVELOPED MATLAB CODE TO COMMUNICATE WITH KVH DSP-300 FIBER OPTIC GYRO

A. WINDOWS XP SIMULINK CODE

1. opensim.m

```
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% Simulink Initialization Function used to open the %
% COM port (COM3) for the Fiber Optic Gyro.         %
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

global f;

f                = serial('COM3');
f.BaudRate       = 38400;

% Ensures at least one full data is stored in the buffer.
f.InputBufferSize = 44;

f.OutputBufferSize = 1;
fopen(f);
```

2. closesim.m

```
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% SIMULINK Function used at the end of the simulation %
% to close the COM port used by the Fiber Optic Gyro. %
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

global f
fclose(f)
clear f
```

3. readsim.m

```
function out = readsim(x)

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%Function to read ASCII data from the KVH DSP-300 Fiber Optic Gyro    %
%Ensure that the serial port is already initialized with 'opensim.m'. %
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
global f last
out=0;

%Ensures buffer is full and reads new data.
%If buffer not full, outputs last good data.

k=f.BytesAvailable;
if k==44
    data = fread(f,k);
    last = data;
else
    data = last;
end

%Searches thru data to find the rate information.
%Converts ASCII rate info to double.

for t=1:27
    test=[data(t:t+6)'];
    if all(test==[49 13 10 32 32 32 32])
        rate=str2num(strcat(char(data(t+7:t+17))'));
    end
end

out=rate;
```

B. XPC TARGET CODE

1. Embedded Matlab Function ‘Parse’

```
function y = fcn(u)

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% Function to parse the ASCII data from the                           %
% KVH DSP-300 Fiber Optic Gyro data stream.                           %
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

% Establishes variable class and size.
d = double(u);
test = [0 0 0 0 0];
rate = [0 0 0 0 0 0 0 0 0 0 0];

%Parses out ASCII rate information from data stream.
i=1;
for i=1:29
    test = [d(i) d(i+1) d(i+2) d(i+3) d(i+4)];
    if all(test==[10 32 32 32 32])
        rate = [d(i+5) d(i+6) d(i+7) d(i+8) d(i+9) d(i+10) d(i+11)...
               d(i+12) d(i+13) d(i+14) d(i+15)];
    end
end

%Converts output to unsigned 8 bit integer for ASCII Decode Block.
y=uint8(rate');
```

APPENDIX F. DEVELOPED MATLAB CODE TO COMMUNICATE WITH MAXON MOTOR EPOS 24/1 POSITIONING CONTROLLER

1. Parse/Decode Embedded MATLAB Function

```
function y = fcn(u)

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%  Embedded MATLAB Function the parses out and decodes  %
%  motor position information from the EPOS 24/1        %
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

%Declares variable sizes and types for emmbedded funct.
d = double(u);
test = [0 0 0 0 0 0 0 0];
encdat = [0 0 0 0];
i=1;

%Searches thru EPOS response to find bytes associated
%with motor position.
for i=1:25
    test = [d(i) d(i+1) d(i+2) d(i+3) d(i+4) d(i+5) d(i+6) d(i+7)];
        %Header right before Position info
    if all(test == [79 79 0 3 0 0 0 0]);
        encdat = [d(i+8) d(i+9) d(i+10) d(i+11)];
    end
end

%Converts position info from hexadecimal format
%to decimal in number of encoder counts.
pos = encdat(4)*16777216+encdat(3)*65536+encdat(2)*256+encdat(1);

%Position info is two's compliment format.
%This checks for negative postion and converts
%encoder counts to degrees.
if pos > 2147483648
    pos=(pos-4294967296)*(360/172032);
else
    pos=pos*(360/172032);
end

%This modulates position about 360
%i.e. 720 degrees == 360.
q=fix(pos/360);
if abs(q)>=1
    pos=pos-q*360;
end

y=pos;
```

2. Initialization Command Sequence

Decimal Command	Purpose
17	Opcode
[3,64,96,0,1,128,0,0,0,57,248]	Clear Faults
79	OK
79	OK
16	Opcode
[1,65,96,0,1,171,240]	Prepare Enable
79	OK
79	OK
17	Opcode
[3,64,96,0,1,6,0,0,0,195,113]	Enable 1
79	OK
79	OK
17	Opcode
[1,65,96,0,1,171,240]	Prepare Enable
79	OK
79	OK
17	Opcode
[3,64,96,0,1,15,0,0,0,82,239]	Enable 2
79	OK
79	OK
17	Opcode
[1,65,96,0,1,171,240]	Prepare Enable
79	OK
79	OK
17	Opcode
[3,64,96,0,1,15,1,0,0,230,135]	Enable 3
79	OK
79	OK
17	Opcode
[1,65,96,0,1,171,240]	Finish Enable
79	OK
79	OK
17	Opcode
[3,246,96,1,1,209,1,0,0,115,30]	Set Pos P Gain
79	OK
79	OK
17	Opcode
[3,246,96,2,1,253,1,0,0,6,3]	Set Pos I Gain
79	OK
79	OK
17	Opcode
[3,249,96,1,1,204,5,0,0,89,95]	Set Vel P Gain
79	OK
79	OK
17	Opcode
[3,249,96,2,1,110,1,0,0,134,168]	Set Vel I Gain
79	OK
79	OK
17	Opcode
[3,96,96,0,1,254,0,0,0,198,85]	Set Velocity Mode
79	OK
79	OK

3. Normal Command Sequence

Decimal Command	Purpose
17	Opcode
Generated by Embedded MATLAB Func	Velocity Command
79	OK
79	OK
17	Opcode
[1,100,96,0,1,157,157]	Report Position Command
79	OK
79	OK

4. Velocity Command Embedded MATLAB Function

```
function y = fcn(u)

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% This Embedded MATLAB Function receives a desired    %
% velocity signal (degs/sec) and encodes a velocity   %
% command for the EPOS 24/1 Positioning Controllel.   %
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

% Initializes variable sizes and types
vel1=u;
crcn = [0 0];
vpn = [0 0];
vpn2 = [0 0];

% Converts velocity (degs/s) to two's complimnetary
% least significant byte leading decimal number of
% representing RPMs.
vel = floor(vel1*60*(16000/84)/360);
if vel < 0
    vel=4294967296+vel;
end
vel2=uint32(vel);
vpp = bitand(vel2,65535);
vpp2 = bitshift(vel2,-16);
vp = double(vpp);
vp2 = double(vpp2);
vpn(2) = floor(vp/256);
vpn(1) = vp-vpn(2)*256;
vpn2(2) = floor(vp2/256);
vpn2(1) = vp2-vpn2(2)*256;

% Concatenates the bytes representing the desired RPMs
% with the velocity command header.
dat = [3,107,32,0,1,vpn,vpn2];

% Calculates the CRC for the given velocity command
m = 2;
d = dat;
```

```
l = d(1) + 1;
d(1) = 256*17+d(1);
i=2;
crc2=0;
for i = 2:l+1
    d(i) = d(m+1)*256+d(m);
    m = m+2;
end
d(l+2) = 0;
m=l+2;
crc     = uint16(0);
for i = 1:m
     shifter = uint16(32768);
     c   = uint16(d(i));
     while all(shifter)
         carry = bitand(crc,uint16(32768));
         crc = uint16(bitshift(crc,1));
         if all(bitand(c,shifter)), crc = crc + 1; end
         if all(carry), crc = bitxor(crc,uint16(4129)); end
         shifter = uint16(bitshift(shifter,-1));
     end
end
crc2=double(crc);
crcn(2) = floor(crc2/256);
crcn(1) = crc2-crcn(2)*256;

% Concatenates the CRC bytes with the velocity
% command  and returns the complete 11 byte command.
y = [dat,crcn]';
```

APPENDIX G. DEVELOPED MATLAB CODE FOR CONTROLLER AND ACTUATOR MAPPING ALGORITHMS

```
function [F1c,a1c,F2c,a2c,Tcmg]= fcn(Fx,Fy,T,theta,del)

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% This Embedded MATLAB Function receives required absolute  %
% force information, attitude, and MSGCMG position and      %
% develops required actuator outputs in the form of dual    %
% vectorable thruster force and position and torque output  %
% for the MSGCMG.                                           %
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%



% Established max thruster force and max differential angle
% between the thrusters.

maxF=.28;
maxalpha=35*pi/180;

% Modulates current attitude about 360 degs (2 Pi).
m=fix(theta/(2*pi));
theta=theta-m*2*pi;

% Converts the required absolute forces (Fx & Fy) into a
% RELATIVE force vector of Fc and ac.
F=[Fx,Fy];
Fc=norm(F);
ac=atan2(Fy,Fx)-theta;

% Sends required torque signal directly to MSGCMG logic
Tcmg=T;

% Generates angle differential (alpha) between the dual
% vectorable thrusters and torque (Tc) based on MSGCMG pos.
Tc=del*.01;
alpha=exp(abs(Tc))*sign(Tc);
if abs(alpha)>maxalpha;
    alpha=maxalpha*sign(Tc);
end

% Determines how to apply alpha and Tc based
% on thruster position
if ac<0 && ac>-pi;
    F1=.5*Fc-Tc;
    F2=.5*Fc+Tc;
else
    F1=.5*Fc+Tc;
    F2=.5*Fc-Tc;
end
if ac>-pi/2 && ac<pi/2
    a1=ac+alpha;
    a2=ac-alpha;
else
    a1=ac-alpha;
    a2=ac+alpha;
end

% Converts negative forces to positive forces
if F1<0
    a1c=-a1;
    F1c=-F1;
else
    a1c=a1;
    F1c=F1;
end
if F2<0
    a2c=-a2;
    F2c=-F2;
else
    a2c=a2;
    F2c=F2;
end

% Converts thruster 1 from a force to a thrust.
% Thruster 2 is converted by physical orientation.
a1c=a1c+pi;

% Ensures that the commanded thruster position
% is between +/- 180 degs (+/- pi).
if a1c>pi
    a1c=a1c-2*pi;
elseif a1c<-pi
    a1c=a1c+2*pi;
else
    a1c=a1c;
end
if a2c>pi
    a2c=a2c-2*pi;
elseif a2c<-pi
    a2c=a2c+2*pi;
else
    a2c=a2c;
end
```

LIST OF REFERENCES

1. DARPA. (2006). Fractionated Spacecraft Workshop Visions & Objectives.

2. Mathieu, C., & Weigel, A.L. (2005). Assessing the Flexibility Provided by Fractionated Spacecraft. *AIAA Space 2005 Conference*.

3. MIT. (2006). *SPHERES.* http://ssl.mit.edu/spheres/.

4. Regehr, M., Acihese, A., Ahmed, A., Aung, M., Bailey, R., Bushnell, C., Clark, K., Hicke, A., Lytlel, B., MacNeal, P., Rasmussen, R., Shields, J., & Singh, G. (2004). The Formation Control Testbed. *Proc 2004 IEEE Aerospace Conference*. Big Sky, MT.

5. Romano, M., Friedman, D., & Shay, Tracy. (2005) Laboratory Experimentation of Autonomous Spacecraft Approach and Docking to a Collaborative Target. Accepted for publication. To Appear AIAA Journal of Spacecraft and Rockets.

6. Friedman, D. (2005). Laboratory Experimentation of Autonomous Spacecraft Docking Using Cooperative Vision Navigation. MS Thesis, Dept. of Astro. And Mech. Eng., NPS.

7. Shay, T.J. (2005). Design and Fabrication of Planar Autonomous Spacecraft Simulator with Docking and Fluid Transfer Capability. MS Thesis, Dept. of Astro. and Mech. Eng., NPS.

8. Hall, Jason. (2006). Design And Integration Of A Three Degrees-Of-Freedom Robotic Vehicle With Control Moment Gyro For The Autonomous Multi-Agent Physically Interacting Spacecraft (AMPHIS) Testbed. MS Thesis, Dept. of Astro. and Mech. Eng., NPS.

9. Eikenberry, Blake. (2006). Guidance And Navigation Software Architecture Design For The Autonomous Multi-Agent Physically Interacting Spacecraft (AMPHIS) Test Bed. MS Thesis, Dept. of Astro. and Mech. Eng., NPS.

10. Ultralife. (2003). UBI-2590 (Part No. UBBL02) Battery Specification.

11. Versalogic. (2005). EPM-CPU-10 Reference Manual.

12. Diamond Systems. (2005). DMM-32X-AT 16-Bit Analog I/O PC/104 Module with Autocalibration User Manual Revision 1.0.

13. RTD. (2005). DMR24 Mechanical Relay Output Board User’s Manual.

14. KVH. (2005). DSP-3000 Fiber Optic Gyro Technical Manual.

15. Crossbow. (2006). TG Series Datasheet.

16. Arc Second. (2005). *Indoor GPS Workspace, Advanced Metrology Edition, User's Guide Version 6.0.*

17. Predyne. (2006). E-EH Series Product Details. http://www.predyne.com/series/series_e-eh.asp.

18. Maxon Motor (2004). *EPOS Positioning Controller Communication Guide*

19. Wie, B. (1998). Space Vehicle Dynamics and Control. *AIAA Educational Series*. Tempe, AZ.

20. SICK. (2004). *LD Laser Scanners Product Information*

INITIAL DISTRIBUTION LIST

1. Defense Technical Information Center
   Ft. Belvoir, Virginia

2. Dudley Knox Library
   Naval Postgraduate School
   Monterey, California

3. Marcello Romano
   Naval Postgraduate School
   Monterey, CA